NEC
NECソリューションイノベータ

InfoCage FileShell SDK
利用ガイド

InfoCage FileShell SDK
Version 6.3
利用ガイド
（0630D02）

InfoCage

はじめに

このたびは、NEC ソリューションイノベータ株式会社の InfoCage FileShell をお買い求めいただき誠にありがとうございます。

InfoCage FileShell は、電子ファイル自身にセキュリティ情報を持たせた暗号化をおこなうことで、

利用者の操作性を損なうことなく重要な情報を永続的に保護する機密情報保護ソフトウェアです。

ご使用になる前に本書をよくお読みになり、製品の取り扱いを十分にご理解ください。

■ 商標について

- Microsoft および Windows は米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
- InfoCage は NEC ソリューションイノベータ株式会社の登録商標です。
- その他、本書に記載されている会社名、商品名は各社の登録商標または商標です。

■ 免責事項

本書および本システムは、ライセンス契約に基づいて使用することができます。
ライセンス契約で明示的に定められていないかぎり、NEC ソリューションイノベータ株式会社は製品、およびその関連文書について、明示的にも暗黙的にも、商品性に関する保証、特定目的への適合性に関する保証、取り扱い、使用、または取引行為に伴う保証について一切の責任を負いません。

本書中のサンプル画面で使用している名称は、すべて架空のものです。実在する品名、団体名、個人名とは一切関係ありません。

本書について

本書は本製品を正しく運用し、効果的に活用するための手引きです。運用を開始する前や運用中に、機能・操作を
確認するためにご利用ください。

本書は、InfoCage FileShell SDK を使ったアプリケーションの開発者、および、InfoCage FileShell SDK を使ったシステムの運用者を対象としています。

ご注意： 本書の一部、または全部を流用・複写することはできません。

本書中の記号について

本書中では、以下の記号を使用しています。これらの記号の意味を正しくご理解になり、
本書をお読みください。

記　号	説　　明
Notice	システムの取り扱いで守らなければならない事柄や特に注意すべき点、確認すべき点を説明します。
参照	関連する内容が記載されているページを紹介しています。
Operation	操作手順を示します。

参考資料について

本書中では、参考資料として以下のガイドを参照するように説明しています。

項　目	ガ　イ　ド　名	番 号
管理者ガイド	InfoCage FileShell 管理者ガイド	0630Knn
NEC File Protection Edition 管理者ガイド	InfoCage FileShell NEC File Protection Edition 管理者ガイド	0630RKnn

＊　末尾の「nn」には、「01」、「02」などの数字(版数)が入ります。
　　版数は、プログラムやマニュアルに変更があった場合に更新されます。

用語の定義

本書では、システム操作の説明に以下のような用語を用いています。本書を確認するにあたって前提としてご理解ください。

用　語	説　　明
FileShell SDK	本ソフトウェア「InfoCage FileShell SDK」のことです。FileShell SDK はファイルの保護、および保護解除をするためのインタフェースを提供します。
MIP	Microsoft Information Protection の略称で、Microsoft 社が提供する情報保護ソリューションのことです。本書では MIP と記載します。 Microsoft Information Protection の詳細については、Microsoft 社の Web サイト等をご参照ください。
ラベル	MIP にて、文書に付与される情報です。 ファイルの保護(暗号化)がおこなえるラベルには権利ポリシーが含まれています。ファイルの分類／保護に使用します。
分類	ファイルにラベルを付与することです。
NEC File Protection Edition(NFP)	オンプレミスまたはクラウドの RMS 基盤を使用せず、NEC 独自の方式により「ファイルの保護」や「利用の制限(権限による制御)」などをおこなう仕組みのことです。
OfficeIRM 形式	RMS の権利ポリシーテンプレートを使用して出力する、Office アプリケーションでサポートしている保護形式です。
Microsoft 互換形式	Microsoft Office ファイル、および PDF ファイルをラベルを用いて出力した際の形式です。Microsoft 互換形式で出力したファイルには、ラベルが付与されます。また、ファイルの保護(暗号化)がおこなえるラベルを用いて出力したファイルは、ラベルの付与と同時に保護されます。
PDFv2 形式	PDF ファイルをラベルを用いて出力した際の保護形式です。
マルチデバイス形式	Azure Information Protection 統合ラベルクライアントで参照可能な形式です。 Azure Information Protection 統合ラベルクライアントの詳細については、Microsoft 社の Web サイト等をご参照ください。
FileShell 形式	RMS の権利ポリシーテンプレートを使用して出力(保護)する FileShell 独自の保護形式です。
FileShell(ラベル)形式	ラベルを使用して出力(分類/保護)する FileShell 独自の保護形式です。
NFP 形式	RMS を使用しない NEC 独自の保護形式です。 「NFP 権利ポリシー認証機能」が利用できるサーバー認証版と、サーバーレスでも利用可能な鍵配布版の 2 種類があります。

用　語	説　　明
オンプレミス RMS	RMS サーバーを自社に構築する運用方法です。Microsoft 社の Active Directory Rights Management Services を利用します。
Azure RMS	RMS サーバーを自社に構築せず、クラウドサービスを利用する運用方法です。Microsoft 社の Azure Rights Management を利用します。 Azure Rights Management の詳細については、Microsoft 社の Web サイト等をご参照ください。
NFP 権利ポリシー / NFP 権利ポリシーテンプレート	NFP で、ファイルを保護/保護解除するために使用する「共通鍵」やファイル利用時の「権限」および有効期限の設定が埋め込まれた情報のことです。

目 次

第 1 章　FileShell SDK について 1

1.1　特長 1

1.2　機能 1

1.3　動作環境について 3

第 2 章　注意事項 4

2.1　運用上の注意事項 4

第 3 章　アプリケーションの開発 9

3.1　開発環境への展開 9

3.2　使用方法 9

3.3　API リファレンス 10

3.3.1　FileShell SDK の初期化処理 － IcfsInitialize() 10

3.3.2　ファイル保護（Office IRM/FileShell、Microsoft 互換/FileShell(ラベル)、NFP 形式）－ IcfsProtectEx() 11

3.3.3　ファイル保護（マルチデバイス形式）－ IcfsProtectMultiDeviceFormat() 12

3.3.4　ファイル分類/保護解除（Office IRM/FileShell、Microsoft 互換/FileShell(ラベル)、NFP 形式）－ IcfsUnprotectEx() 13

3.3.5　ファイル保護解除（マルチデバイス形式）－ IcfsUnprotectMultiDeviceFormat() 14

3.3.6　メモリ解放 － IcfsFreeMemory() 15

3.3.7　分類／保護状態確認 － IcfsIsProtectedEx() 16

3.3.8　保護形式確認 － IcfsIsMultiDeviceFormatFile() 17

3.3.9　分類状態確認 － IcfsIsLabeled() 18

3.3.10　コンテンツ ID 取得 － IcfsGetContentIdEx() 19

3.3.11　権利ポリシー作成 － IcfsCreateRightsPolicyFile() 20

3.3.12　ファイルフォーマット取得 － IcfsGetFileFormat() 23

3.3.13　FileShell SDK の終了処理 － IcfsUninitialize() 24

3.4　エラーコード 25

3.5　サンプルコード 28

第 4 章　FileShell SDK 運用環境構築 33

4.1　運用環境構築の流れ 33

4.2　必要なソフトウェアのインストール 33

4.3　RMS サーバーへの接続に必要な設定 34

4.3.1　RMS 証明パイプラインに権限を追加する 34

4.3.2　信頼された発行ドメインの設定 36

4.4　インストーラー作成時に設定する情報の取得 37

4.4.1　MIP/Azure RMS を利用する場合に必要な情報の設定・取得 37

4.4.2 NFP の緊急保護解除用の公開鍵の取得 45
4.5 FileShell SDK のインストール 48
4.5.1 インストールパッケージの展開 48
4.5.2 インストーラーの作成 49
4.5.3 インストール 56
4.5.4 MIP による分類/保護や解除をおこなう場合の設定 58
4.6 FileShell SDK の環境設定 59
4.6.1 RMS 認証情報の設定 59
4.6.2 ログ出力情報の設定 59
4.7 権利ポリシーテンプレートの準備 60
4.7.1 オンプレミス RMS サーバー上の権利ポリシーテンプレートの取得と保存 60
4.7.2 Azure RMS サーバー上の権利ポリシーテンプレートの取得と保存 61
4.7.3 NFP 権利ポリシーテンプレートの取得と保存 63
4.8 Azure RMS 権利ポリシーテンプレートの編集 65
4.9 権利ポリシーテンプレートのインポート 67
4.9.1 Office IRM/FileShell 形式で保護する場合 67
4.9.2 NFP 形式で保護する場合 70
4.10 ラベル ID の取得 71
4.11 ラベルへの保護解除権限の付与 73

第 5 章 環境設定ツール 77
5.1 機能一覧 77
5.2 起動方法 77
5.3 環境設定ツールの操作 77
5.3.1 RMS 認証情報設定 77
5.3.2 権利ポリシーテンプレートの管理 80
5.3.3 ログ出力情報設定 82
5.3.4 オプション設定 83
5.3.5 動作環境表示 85
5.3.6 設定情報ファイル出力 85

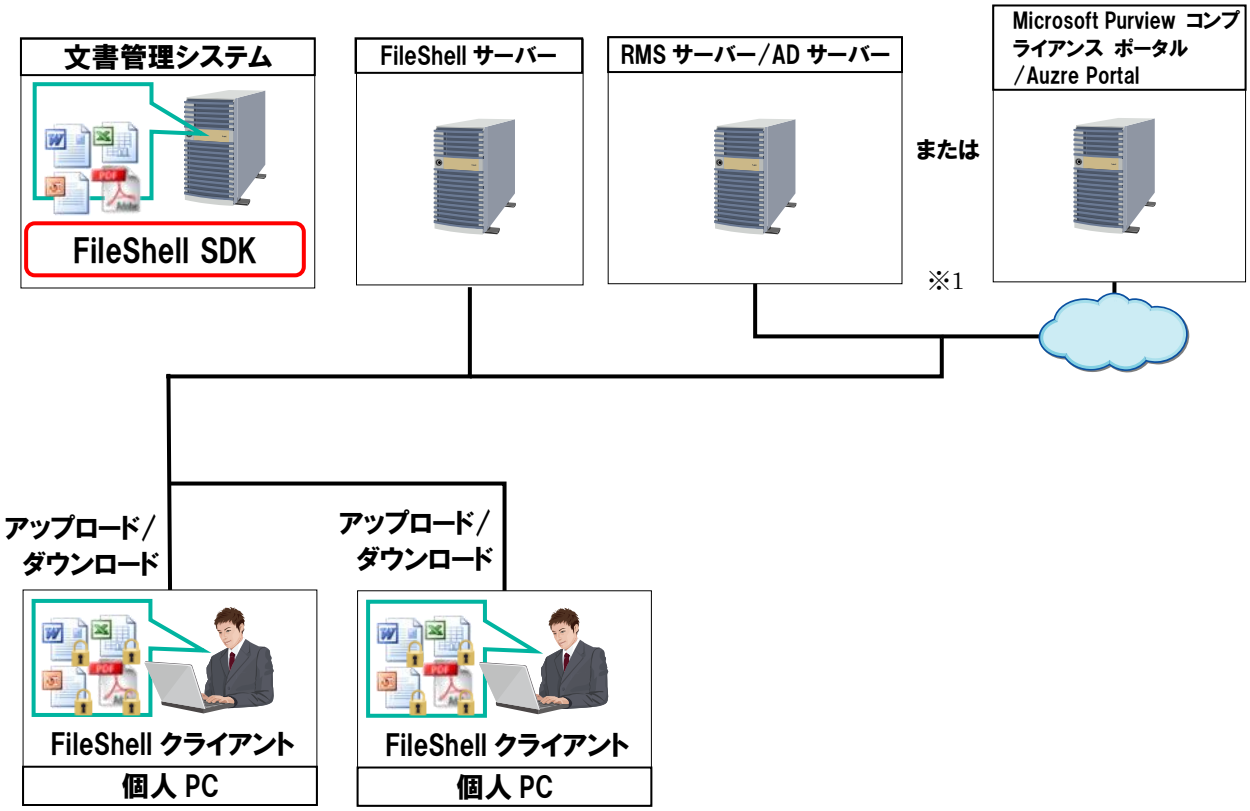
第 6 章 バージョンアップインストール 86
6.1 V2.0 以上からのバージョンアップ 86
6.2 V2.0 未満からのバージョンアップ 86

第 7 章 アンインストール 88

第1章 FileShell SDK について

1.1 特長

FileShell SDK はファイルの保護、および保護解除をするためのインタフェースを提供します。開発者は、FileShell SDK を利用して、たとえば、文書管理システム上の操作と連動して、ファイルダウンロード時に対象ファイルを保護、アップロード時に保護解除するようなアプリケーションを開発することが可能となります。



＊ (※1) OfficeIRM 形式、マルチデバイス形式、FileShell 形式、および Microsoft 互換/FileShell(ラベル)形式での保護を利用する場合、FileShell SDK の動作には、RMS サーバー/AD サーバーまたは Microsoft Purview コンプライアンス ポータル/Auzre Portal の環境が必要です。

＊ FileShell SDK は、FileShell サーバーを利用しないため、FileShell サーバーとの通信は発生しません。

1.2 機能

FileShell SDK は、以下の機能を提供します。各形式ごとに使用する機能は以下のとおりです。

OfficeIRM/FileShell 形式、Microsoft 互換/FileShell(ラベル)形式

機能名	概要	関数インタフェース名	参照
ファイル分類/保護	指定されたファイルを分類/保護します。	IcfsProtectEx()	3.3.2
ファイル分類/保護解除	指定された分類/保護ファイルの適用を解除します。	IcfsUnprotectEx()	3.3.4

分類/保護状態確認	指定されたファイルが分類/保護済みかどうかの確認をおこないます。	IcfsIsProtectedEx()	3.3.7
分類状態確認	指定されたファイルが分類(ラベル付与のみ)済みかどうかの確認をおこないます。	IcfsIsLabeled	3.3.9
コンテンツ ID 取得	指定された分類/保護ファイルのコンテンツ ID※1を取得します。	IcfsGetContentIdEx()	3.3.10
権利ポリシー作成	指定した権限に従って、ローカル権利ポリシーファイル(xml)を生成します。	IcfsCreateRightsPolicyFile ()	3.3.11
ファイルフォーマット取得	分類/保護されたファイルの保護形式を取得します。	IcfsGetFileFormat()	3.3.12

マルチデバイス形式

機能名	概要	関数インタフェース名	参照
ファイル保護	指定されたファイルをマルチデバイス形式で保護します。	IcfsProtectMultiDeviceFormat()	3.3.3
ファイル保護解除	指定されたマルチデバイス形式の保護ファイルの保護を解除します。	IcfsUnprotectMultiDeviceFormat()	3.3.5
メモリ解放	マルチデバイス形式でファイルを保護または保護解除する際に確保したメモリを解放します。	IcfsFreeMemory()	3.3.6
保護状態確認	指定されたファイルが保護済みかどうかの確認処理をおこないます。	IcfsIsProtectedEx()	3.3.7
保護形式確認	ファイルがマルチデバイス形式で保護済みかどうかの確認処理をおこないます。	IcfsIsMultiDeviceFormatFile()	3.3.8
コンテンツ ID 取得	指定された保護ファイルのコンテンツ ID※1を取得します。	IcfsGetContentIdEx()	3.3.9
ファイルフォーマット取得	保護されたファイルの保護形式を取得します。	IcfsGetFileFormat()	3.3.12

NFP 形式

機能名	概要	関数インタフェース名	参照
ファイル保護	指定されたファイルを保護します。	IcfsProtectEx()	3.3.2
ファイル保護解除	指定された保護ファイルの保護を解除します。	IcfsUnprotectEx()	3.3.4
保護状態確認	指定されたファイルが保護済みかどうかの確認をおこないます。	IcfsIsProtectedEx()	3.3.7
コンテンツ ID 取得	指定された保護ファイルのコンテンツ ID※1を取得します。	IcfsGetContentIdEx()	3.3.9
ファイルフォーマット取得	保護されたファイルの保護形式を取得します。	IcfsGetFileFormat()	3.3.12

(※1) コンテンツ ID:ファイルを保護した際に一意に設定される識別コードです。FileShell SDK を利用するアプリケーション側で、コンテンツ ID を利用して保護ファイルを識別することが可能です。

1.3　動作環境について

FileShell SDK の動作環境は以下のとおりです。

■ FileShell SDK　動作環境

ハードウェア	
CPU	2.0GHz 相当以上の x86 互換アーキテクチャのプロセッサ
メモリ	2GB 以上
ハードディスク	FileShell SDK のインストールに 40MB 以上の空き容量が必要です。 運用時には、ポリシー数やログ量に応じた空き容量が必要です。
ネットワークインタフェース	100Mbps 以上(IPv4 のみ)
ソフトウェア	
オペレーティングシステム(*1)(*2)	Microsoft Windows Server 2022 Standard (64bit) Microsoft Windows Server 2019 Standard (64bit) Microsoft Windows Server 2016 Standard (64bit)
その他	Microsoft .NET Framework 4.7.2 (*3) Microsoft .NET Framework 4.7.2 日本語 Language Pack(日本語 OS のみ)(*3) RMS Client V2.1(*4) (*5) Visual Studio 2015、2017、2019、および 2022 用 Visual C++再頒布可能パッケージ (*5)
その他、必要環境	
FileShell 製品、その他	FileShell SDK を動作させるためには、RMS サーバー、AD サーバーの運用環境が必要です。

(*1)　Microsoft 社から提供される最新セキュリティパッチの適用をお奨めします。

(*2)　対象言語は、日本語・英語です。

(*3)　Windows Server 2016 をご利用の場合は、後述(*5)の「Microsoft ダウンロードセンター」よりダウンロードしてください。Windows Server 2022 および 2019 には、Microsoft .NET Framework 4.7.x が OS のデフォルトでインストールされています。動作に必要となるため、削除しないでください。

(*4)　保護や保護解除にオンプレミスの AD RMS を利用する場合、および Ver6.0 未満のバージョンの FileShell クライアントで保護したファイルを保護解除する場合に必要です。

(*5)　「Microsoft ダウンロードセンター」よりダウンロードする必要があります。
以下の Web サイトよりファイルをダウンロードしてから、インストールしてください。(2023/12/15 時点)

タイトル、URL	備考
Visual Studio 2015、2017、2019、および 2022 用 Visual C++再頒布可能パッケージ https://docs.microsoft.com/ja-JP/cpp/windows/latest-supported-vc-redist?view=msvc-170	バージョン 14.28.29325 以降を使用してください。上記より古いバージョンを使用した場合、インストールに失敗する場合があります。
Microsoft .NET Framework 4.7.2 https://dotnet.microsoft.com/ja-jp/download/dotnet-framework/net472	各言語用の Language Pack も本ページ内のリンクからダウンロードできます。
RMS Client V2.1 https://www.microsoft.com/ja-jp/download/details.aspx?id=38396	-

■ 連携可能な InfoCage FileShell　製品バージョン

対応バージョン	
FileShell クライアント	V1.1、V2.0、V2.1、V3.0、V3.1、V3.2、V4.0、V4.1、V5.0、V6.0、V6.1、V6.2、V6.3

第2章 注意事項

2.1 運用上の注意事項

＊ 本ソフトウェアのインストーラーは、必ずインストーラ作成支援ツールを使用して作成してください。テキストエディタ等によるインストーラー設定ファイルの編集はおこなえません。

＊ FileShell SDK V6.3 より、FileShell SDK を利用して作成するアプリケーションの開始時と終了時に、それぞれ本 SDK の API による初期化処理、および終了処理をおこなう必要があります。これにともない V6.3 未満の環境からアップデートする場合は、V6.3 未満の FileShell SDK で作成したアプリケーションに対して、これらの処理を追加して更新する必要があります。詳しくは「第 6 章 バージョンアップインストール」を参照してください。
FileShell SDK V6.3 環境下で上記の初期化、および終了処理を追加していないアプリケーションを実行すると、アプリケーションがフリーズするなどの事象が発生する場合があります。アプリケーションを更新しない場合は、V6.2 以下の FileShell SDK をご利用ください。
なお、FileShell SDK V6.3 を利用しない環境では以下の事象が発生する場合があります。
- AES 256-CBC で保護された、Office ファイルが保護解除できない。
- 「5.3.1 RMS 認証情報設定」の[MSAL 設定]に設定情報がない場合に、PDFV2 形式ファイルが二重保護される。

＊ FileShell SDK V6.1 より、Azure RMS を利用して保護/保護解除をおこなう場合、対称鍵(Symmetric Key) による認証はできなくなりました。これにともない、FileShell SDK V6.1 未満の環境からアップデートする場合は、Azure RMS の認証をクライアントシークレットによる認証に変更する必要があります。
FileShell SDK のバージョンアップをおこなう前に、「4.4.1 MIP/Azure RMS を利用する場合に必要な情報の設定・取得」に記載の手順で、Azure Portal にてアプリケーションを登録し、クライアントシークレットの作成をおこなってください。

＊ MIP、もしくは Azure RMS を利用して保護/保護解除をおこなう場合に必要となるクライアントシークレットは、有効期限が切れた場合、Azure RMS の認証がおこなえなくなるため、運用年数、セキュリティリスクなどを考慮の上、適切な期限を設定し、運用中に期限が切れることのないよう管理をおこなってください。
クライアントシークレットの有効期限は、最大 24 か月です。
有効期限が切れた場合は、「4.4.1.3 クライアントシークレットの作成」に記載の手順でクライアントシークレットを再作成し、FileShell SDK 環境設定ツールに適用しなおしてください。
なお、クライアントシークレットが漏えいすると、Azure Active Diretory からユーザー・グループ・AU などの情報を第三者に取得される可能性があります。クライアントシークレットの管理には十分ご注意ください(漏えいの疑いがある場合は、すぐに Azure Portal でアプリの設定を無効化するなど、処置をおこなってください)。

＊ FileShell SDK では、FileShell SDK を利用するアプリケーションの実行アカウントを使ってファイルの分類/保護、およびそれらの適用解除などの処理をおこなっています。このため、FileShell SDK を利用するアプリケーションの実行時には、以下の注意事項があります。
- オンプレミスの RMS サーバーと連携して FileShell SDK を利用する場合、以下のいずれかのアカウントで実行されているアプリケーションからのみ利用可能です。
  - Active Directory のドメインアカウント
  - System アカウント
  - Network Service アカウント

＊ 本ソフトウェアで分類(ラベルの付与)、保護(暗号化)、およびそれらの適用を解除するファイルに対して、前述したアカウントが以下のアクセス権限を有している必要があります。
  - 読み取り

- 書き込み
- 読み取りと実行
- 変更
- フォルダーの内容の一覧表示
- 所有権の取得
- アクセス許可の読み取り
- アクセス許可の変更

＊ 保護および保護解除後のファイルは、ファイルシステム上では、ファイルが存在しているフォルダーと同じアクセス権限になります。

＊ FileShell SDK を利用してファイルの分類/保護の適用解除をおこなう場合、実行アカウントにはその保護ファイルに対するフルコントロール権限が必要となります。そのため、FileShell SDK で分類/保護したファイルへの適用解除できるようにするためには以下のようにします。

【オンプレミスの RMS 環境の場合】

①ActiveDirectory にて、ファイルの保護/保護解除をおこなうための専用のセキュリティグループと専用のアカウントを作成し、セキュリティグループにアカウント※1を所属させる。

(※1) System アカウント、または、Network Service アカウントで実行する場合は、FileShell SDK を実行するマシンのコンピューター アカウントを、セキュリティグループに所属させてください。

例） 動作するマシン名が ”apserver1” の場合、 コンピューター アカウント “apserver1” をセキュリティグループに所属させます。

②保護時に利用する権利ポリシーテンプレートには、このセキュリティグループに対してフルコントロール権限を付与する。

参照 権利ポリシーテンプレートについては、「4.7 権利ポリシーテンプレートの準備」を参照してください。

③FileShell SDK を利用するアプリケーションをこの専用アカウントで実行する。

【Azure RMS 環境の場合】

「4.8 Azure RMS 権利ポリシーテンプレートの編集」を参照してください。

＊ 本バージョンの FileShell SDK では、V1.1.5 未満の FileShell SDK 用に作成したアプリケーションは動作しません。

＊ 本バージョンの FileShell SDK では、V1.1.5 未満の FileShell SDK と同じマシン上にインストールして動作させることはできません。

＊ FileShell SDK は、FileShell の別コンポーネント(FileShell サーバー、FileShell クライアント、FileShell プロテクタ for Microsoft SharePoint Server)と同じマシン上にインストールしないようにしてください。互いに正常動作しなくなる可能性があります。

＊ オンプレミスの AD RMS サーバーを SSL で構築する場合、AD RMS のサーバー証明書は、自己署名証明書ではなく、信頼されたルート証明機関から発行された証明書を使用することを推奨します。評価環境等で自己署名証明書を使用する場合、FileShell SDK を利用するためには、FileShell SDK が動作するマシンのローカルコンピューターの信頼されたルート証明機関に、AD RMS サーバーのサーバー証明書（自己署名証明書）をインストールする必要があります。

＊ Windows Server 2012 以降で、IIS の Web アプリケーションのアプリケーションプール ID にカスタムアカウント(ドメインアカウント)を指定して実行されているアプリケーションから、FileShell SDK を実行した場合に、RMS のマシン認証に失敗します。本事象は、Windows Server 2012 以降の内部実装の変更によるもので、アプリケーションプール ID を System アカウント(ビルトインアカウント)に変更することで回避可能です。

＊ 権利ポリシーテンプレートを配置した後は、その権利ポリシーでの保護を使用しなくなるまでは、権利ポリシーテンプレートの移動、削除、名前の変更などの操作はおこなわないでください。ファイルの保護は、この

権利ポリシーテンプレートを参照しておこなわれるため、これらの操作をおこなった場合、保護に失敗する場合があります。

＊ 本ソフトウェアでは、権利ポリシーテンプレートの管理はおこないません。
権利ポリシーテンプレートの管理は、RMS サーバーの管理者がおこなってください。

＊ 以下の拡張子をもつファイルは、OfficeIRM 形式で保護した後のファイルサイズが 2.0GB 以上となる場合、OS 仕様により保護に失敗します。

・Microsoft Office 形式のファイル
doc、dot、xla、xls、xlt、pps、ppt、pot、docm、docx、dotm、dotx、xlam、xlsb、xlsm、xlsx、xltm、xltx、xps、potm、potx、ppsx、ppsm、pptm、pptx、thmx

・Microsoft Visio 形式のファイル
vsdx、vstx、vsdm、vstm

＊ FileShell 形式で保護されるファイルについて、ファイルパスが 260 文字以上、かつ、ファイルを保護した後のファイルサイズが 2GB(※)を超えるときは、対象のファイルを保護することができません。ファイルパスが 259 文字以内となる場所にファイルを移動してから保護してください。
(※)保護前のファイルサイズが、およそ 1.8GB 前後とお考えください。

＊ Mircosoft Office 形式の拡張子については、Microsoft Office 形式のファイルに関する注意事項を参照してください。
＊ Microsoft Visio 形式のファイル(vsdx、vstx、vsdm、vstm)を FileShell 形式で保護したファイルは、本注意事項に該当します。

＊ ADRMS/Azure RMS でファイルを分類／保護した際、使用ライセンス(End User License:EUL)と呼ばれるファイルが OS によって作成されます。EUL ファイルはファイル保護がおこなわれるたびに蓄積されていき、ディスク使用量の消費につながるため、これらを利用しない場合には定期的に削除することをおすすめします。
※EUL ファイルは以下のフォルダーに作成されます(「EUL」から始まる、拡張子が「.drm」のファイルが対象です)。
%allusersprofile%¥Microsoft¥MSIPC¥Server¥<SID>

＊ <SID>には、本ソフトウェアを動作させる時に使用するアカウントのものが入ります。
SID の確認は、本ソフトウェアを動作させる時に使用するアカウントでログオンし、コマンドプロンプトから、「whoami /user」コマンドを実行します。
コマンドの詳細については、「whoami /?」を実行してください。

＊ Microsoft Purview コンプライアンス ポータルや Auzre Portal に接続するためにプロキシサーバーの設定が必要なネットワーク環境の場合、保護、および保護解除するアカウントにプロキシサーバーの設定が適用されている必要があります。

詳細は以下のリンク先の情報を参照してください。
(Microsoft 365 Common および Office Online)
https://docs.microsoft.com/ja-jp/microsoft-365/enterprise/urls-and-ip-address-ranges?view=o365-worldwide#microsoft-365-common-and-office-online

(bitsadmin util および setieproxy)
https://docs.microsoft.com/ja-jp/windows-server/administration/windows-commands/bitsadmin-util-and-setieproxy

なお、保護、および保護解除するアカウントにプロキシサーバーの設定を適用するためには、以下の手順をおこなってください(変更をおこなうには管理者権限が必要です)。

[プロキシサーバーの設定手順]

1) コマンドプロンプトを「管理者として実行」で起動します。
2) 以下のコマンドを実行します。
自動構成スクリプトを使用する/しないにより手順が異なります。

● **プロキシサーバーの設定に自動構成スクリプトを使用する場合**

>bitsadmin /util /setieproxy LOCALSYSTEM AUTOSCRIPT <スクリプトの URL>

(例)
>bitsadmin /util /setieproxy LOCALSYSTEM AUTOSCRIPT
http://proxy.server.com/autoproxy.pac

● **プロキシサーバーの設定に自動構成スクリプトを使用しない場合**

>bitsadmin /util /setieproxy LOCALSYSTEM MANUAL_PROXY <プロキシサーバー1>,<プロキシサーバー2>, ... NULL

(例 1) LocalSystem アカウントにプロキシサーバーの設定を適用する
>bitsadmin /util /setieproxy LOCALSYSTEM MANUAL_PROXY 11.22.33.44:8080 NULL

(例 2) LocalSystem アカウントに複数のプロキシサーバーの設定を適用する
>bitsadmin /util /setieproxy LOCALSYSTEM MANUAL_PROXY
proxy.server.com:8080,11.22.33.44 NULL

＊ ″NULL″の前にスペースを入れてください。

● **プロキシサーバーの設定を削除する場合**

>bitsadmin /util /setieproxy LOCALSYSTEM NO_PROXY

＊ FileShell クライアントがバージョン 2.1 未満の場合、FileShell クライアント側でマルチデバイス形式でのファイルを保護または保護解除することはできません。
また、FileShell クライアントがバージョン 4.0 未満の場合、FileShell クライアント側で NFP 形式のファイルを保護または保護解除することはできません。
本ソフトウェアと該当バージョンの FileShell クライアントを連携し運用する場合は、注意が必要です。

＊ マルチデバイス形式保護 API を使用して Office 形式ファイルを保護した場合、Office IRM 形式として保護されます。

＊ パスワードで保護された Office 2007 形式ファイルは、マルチデバイス形式保護 API を使用して保護することはできません。

＊ 本ソフトウェアでは、Azure RMS の「利用状況の確認」機能は使用できません。

＊ FileShell クライアントがバージョン 4.0 未満の場合、FileShell クライアント側で NFP 形式のファイルを閲覧・編集することはできません。

＊ NFP 権利ポリシーを使用して保護できるファイルのサイズは、4GB までとなります。4GB を超えるファイルを保護しようとした場合、エラーとなり保護することはできません。

＊ FileShell クライアントがバージョン 6.0 未満の場合、FileShell クライアント側で Microsoft 互換/FileShell 形式のファイルを閲覧・編集することはできません。

＊ FileShell クライアントがバージョン 6.1 未満の場合、FileShell クライアント側でサーバー認証版の NFP 形式のファイルを閲覧・編集することはできません。

＊ 解除時や分類を変更する際に 理由が求められるラベルの場合、理由は以下の固定文字列となります。
″Change by FileShell″

＊ FileShell SDK でサーバー認証版の NFP 権利ポリシーテンプレートを使用する際は、FileShell サーバーの″バックアップ″機能で出力された NFP 権利ポリシーテンプレートを使用してください。
(FileShell サーバーの″エクスポート″機能で出力されたサーバー認証版の NFP 権利ポリシーでは、保護を解除できません)

＊ Microsoft 互換形式のファイル、および FileShell(ラベル)形式として保護された PDF ファイルについては、分類／保護状態を確認することができません。（保護なしと判定されます。また、保護されたこれらのファイルに対して再度 分類／保護をおこなうと二重に保護されます。これらのファイルを扱う場合は、「5.3 環境設定ツールの操作」の「5.3.1 RMS 認証情報設定」を参照し、MIP/Azure RMS の設定をおこなってください。）

第3章 アプリケーションの開発

本章では、FileShell SDK を使ったアプリケーションの開発方法を紹介します。

3.1 開発環境への展開

アプリケーションの開発環境に、FileShell SDK のヘッダファイルとライブラリファイルを展開します。

Operation

1. FileShell SDK のインストールパッケージを任意のフォルダーに展開してください。

   参照 インストールパッケージの展開方法は「4.5.1 インストールパッケージの展開」の手順 1 を参照してください。

2. Include フォルダー、Lib フォルダーにあるヘッダファイルとライブラリファイルをアプリケーションの開発環境にコピーします。

3.2 使用方法

FileShell SDK を使用するには、以下の 2 種類の方法があります。

- 利用する C/C++アプリケーションよりヘッダファイル(IcfsProtector.h)をインクルードしてコンパイルし、インポートライブラリ(IcfsProtector.lib)をリンクして使用する。
- 利用するアプリケーションより IcfsProtector.dll をロードライブラリで動的にロードして使用する。

Notice

・ヘッダファイル、ライブラリは C/C++の形式です。
・IcfsProtector.dll は FileShell SDK の他のライブラリ(DLL)をリンクして利用しています。そのため、FileShell SDK を利用するアプリケーションは、FileShell SDK がインストールされているフォルダー上で実行するか、「4.5.4 MIP による分類/保護や解除をおこなう場合の設定」の記載を参考に、必要な FileShell SDK のライブラリーをアプリケーションの実行フォルダーにコピーして実行してください。
　＊FileShell SDK は、デフォルトでは以下にインストールされています。
　〈システムドライブ〉¥Program Files¥NEC¥InfoCageFileShellSDK¥

3.3 API リファレンス

FileShell SDK が提供している API について説明します。

＊ FileShell V6.3 より、FileShell SDK で作成するアプリケーションの開始時に「3.3.1 FileShell SDK の初期化処理 ― IcfsInitialize()」、終了時に「3.3.13 FileShell SDK の終了処理 ― IcfsUninitialize()」、をおこなう必要があります。

参照 各出力形式の詳細については、『管理者ガイド』の「FileShell の出力形式について」を参照してください。

3.3.1 FileShell SDK の初期化処理 ― IcfsInitialize()

《概要》

FileShell SDK の初期化処理を実施します。**この処理は FileShell SDK の利用を開始するときに、必ず最初に一度呼び出す必要があります。**

《構文》

HRESULT WINAPI IcfsInitialize();

＊ FileShell SDK の利用を終了する際には、IcfsUninitialize で終了処理をする必要があります。

参照 IcfsUninitialize については、「3.3.13 FileShell SDK の終了処理 ― IcfsUninitialize()」を参照してください。

《引数》

なし

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.2 ファイル保護（Office IRM/FileShell、Microsoft 互換/FileShell(ラベル)、NFP 形式）－ IcfsProtectEx()

《概要》

Office IRM/FileShell 形式、Microsoft 互換/FileShell(ラベル)形式、または NFP 形式でファイルの出力(保護、または分類と保護)をおこないます。ファイルに付与する権限情報は、権利ポリシーテンプレート、またはラベル ID で指定します。

ファイルの保護形式は、pszRptPath に指定する権利ポリシーテンプレート、またはラベル ID によって自動的に判断されます。

参照 権利ポリシーテンプレートの作成方法は「4.7 権利ポリシーテンプレートの準備」を参照してください。ラベル ID の取得方法は「4.10 ラベル ID の取得」を参照してください。

《構文》

```
HRESULT WINAPI IcfsProtectEx(

LPCWSTR   pszFileSrc,      // [in] 分類/保護するファイルパス
LPCWSTR   pszFileDst,      // [in] 分類/保護後の出力ファイルパス
LPCWSTR   pszRptPath       // [in] 付与する権利ポリシーテンプレートファイルパス
                               または分類に使用するラベルのラベル ID（Guid）
);
```

《引数》

pszFileSrc [in]　分類/保護するファイル名(ドライブ名を含む絶対パス)を指定します。

pszFileDst [in]　分類/保護後の出力ファイル名(ドライブ文字を含む絶対パス)を指定します。ここで指定された出力ファイル名で分類/保護ファイルが出力されます。分類/保護するファイル名と、分類/保護後の出力ファイル名を同一にすることはできません。

pszRptPath [in]　Office IRM/FileShell 形式、または NFP 形式で保護する場合
ドライブ文字を含む絶対パスで、権利ポリシーテンプレートファイルを指定します。

指定する権利ポリシーテンプレートのパスは、FileShell SDK 環境設定ツールで権利ポリシーテンプレートをインポートしたあと、同ツールの「パスをコピー」ボタンで取得します。

Microsoft 互換/FileShell(ラベル)形式で分類/保護する場合
分類/保護に使用するラベル ID(Guid) を、"{ }"付きで指定)します。

指定例：
IcfcProtect(pszSrc, pszDst, L"{deb36b0e-9fac-41a5-8c9e-eca546ad3459}");

＊ 適用時にユーザーによるアクセス許可の割り当てがおこなえるラベルは指定できません。

＊ ラベル ID の GUID は 取得時の文字列のまま指定してください。
（大文字などにするとエラーとなります）

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.3 ファイル保護（マルチデバイス形式）－ IcfsProtectMultiDeviceFormat()

《概要》

マルチデバイス形式でファイルの保護をおこないます。ファイルに付与する権限情報は、権利ポリシーテンプレートで指定します。

参照 権利ポリシーテンプレートの作成方法は「4.7 権利ポリシーテンプレートの準備」を参照してください。

《構文》

```
HRESULT WINAPI IcfsProtectMultiDeviceFormat(
LPCWSTR   pszFileSrc,        // [in] 保護するファイルパス
LPCWSTR   pszOutputDir,      // [in] 保護するされたファイルを出力するフォルダーパス
LPCWSTR   *ppszOutputFilePath, // [out] 保護後の出力ファイルパス
LPCWSTR   pszRptPath         // [in] 付与する権利ポリシーテンプレートファイルパス
);
```

＊ ppszOutputFilePath 利用後は、IcfsFreeMemory で解放する必要があります。

参照 IcfsFreeMemory については、「3.3.6 メモリ解放 － IcfsFreeMemory()」を参照してください。

《引数》

pszFileSrc [in]　保護するファイル名(ドライブ名を含む絶対パス)を指定します。

pszOutputDir [in] 保護するされたファイルを出力するフォルダー(ドライブ名を含む絶対パス)を指定します。

ppszOutputFilePath [out]　保護後のファイルパスが返却されます。

pszRptPath [in]　ファイルの権限情報が記述された権利ポリシーテンプレートファイル名(ドライブ文字を含む絶対パス)を指定します。

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.4 ファイル分類/保護解除（Office IRM/FileShell、Microsoft 互換/FileShell(ラベル)、NFP 形式）－ IcfsUnprotectEx()

《概要》

Office IRM/FileShell 形式、Microsoft 互換/FileShell(ラベル)形式、または NFP 形式で分類/保護されているファイルへの適用を解除します。

《構文》

```
HRESULT WINAPI IcfsUnprotectEx(
LPCWSTR    pszFileSrc,        // [in] 分類/保護の適用を解除するファイルパス
LPCWSTR    pszFileDst         // [in] 分類/保護の適用解除後の出力ファイルパス
);
```

《引数》

pszFileSrc [in]　分類/保護解除するファイル名(ドライブ名を含む絶対パス)を指定します。

pszFileDst [in]　分類/保護の適用解除後の出力ファイル名(ドライブ文字を含む絶対パス)を指定します。ここで指定された出力ファイル名で分類/保護の適用解除後のファイルが出力されます。分類/保護の適用を解除するファイル名と分類/保護の適用解除後の出力ファイル名を同一にすることはできません。

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.5 ファイル保護解除（マルチデバイス形式）－ IcfsUnprotectMultiDeviceFormat()

《概要》

マルチデバイス形式形式で保護されているファイルの保護を解除します。

《構文》

HRESULT WINAPI IcfsUnprotectMultiDeviceFormat(

LPCWSTR *pszFileSrc*, // [in] 保護解除するファイルパス
LPCWSTR *pszOutputDir*, // [in] 保護解除されたファイルを出力するフォルダーパス
LPCWSTR **ppszOutputFilePath* // [out] 保護解除後の出力ファイルパス
);

＊ ppszOutputFilePath 利用後は、IcfsFreeMemory で解放する必要があります。

参照 IcfsFreeMemory については、「3.3.6 メモリ解放 － IcfsFreeMemory()」を参照してください。

《引数》

pszFileSrc [in] 保護解除するファイル名(ドライブ名を含む絶対パス)を指定します。

pszOutputDir [in] 保護解除されたファイルを出力するフォルダー(ドライブ名を含む絶対パス)を指定します。

ppszOutputFilePath [out] 保護後のファイルパスが返却されます。

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.6 メモリ解放 ― IcfsFreeMemory()

《概要》

マルチデバイス形式でファイルを保護または保護解除する際、ppszOutputFilePath で確保したメモリを解放します。

＊ FileShell 形式で保護または保護解除する場合は実行不要です。

《構文》

void WINAPI IcfsFreeMemory(

LPCWSTR *pszOutputFilePath* // [in] 解放するメモリのポインタ
);

《引数》

pszOutputFilePath[in] IcfsProtectMultiDeviceFormat または IcfsUnprotectMultiDeviceFormat にて取得したファイルパスのメモリを解放します。

《戻り値》

なし。

3.3.7 分類／保護状態確認 ― IcfsIsProtectedEx()

《概要》

ファイルが権利ポリシーテンプレート、もしくはラベルにより分類/保護済みかどうかの確認処理をおこないます。

《構文》

```
HRESULT WINAPI IcfsIsProtectedEx(
    LPCWSTR     pszFile,        // [in] 確認するファイルパス
    BOOL        *pbRet          // [out] 分類/保護されているかどうか
);
```

《引数》

pszFile [in] 分類／保護状態を確認するファイル名(ドライブ文字を含む絶対パス)を指定します。

pbRet [out] 分類／保護の有無についてチェックした結果が格納されます。
分類／保護されている場合は TRUE、分類／保護されていない場合は FALSE が格納されます。

＊ Microsoft 互換形式で分類(ラベル付与のみ)されているファイルは FALSE となります。

＊ PDFv2 形式で分類/保護されているファイルは FALSE となります。

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.8 保護形式確認 ― IcfsIsMultiDeviceFormatFile()

《概要》

ファイルが分類(ラベル付与)されているかどうかの確認処理をおこないます。

《構文》

```
HRESULT WINAPI IcfsIsMultiDeviceFormatFile(
        LPCWSTR     pszFile,        // [in] 確認するファイルパス
        BOOL        *pbRet          // [out] 保護されているかどうか
);
```

《引数》

pszFile [in] 保護形式を確認するファイル名(ドライブ文字を含む絶対パス)を指定します。

pbRet [out] 保護形式についてチェックした結果が格納されます。
マルチデバイス形式の場合は TRUE、保護されていないまたはマルチデバイス形式でない場合は FALSE が格納されます。

＊ 以下の拡張子については、Microsoft 互換/FileShell(ラベル)形式で保護されたファイルでも、マルチデバイス形式として認識されます。
pfile,ptxt,pxml,pjpg,pjpeg,ppng,ptif,ptiff,pbmp,pgif,pjpe,pjfif,pjt,pjif,pjfi

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.9 分類状態確認 ― IcfsIsLabeled()

《概要》

ファイルが分類(ラベル付与)されているかどうかの確認処理をおこないます。

《構文》

```
HRESULT WINAPI IcfsIsLabeled(
        LPCWSTR     pszFile,        // [in] 確認するファイルパス
        BOOL        *pbRet          // [out] 分類(ラベル付与)されているかどうか
);
```

《引数》

pszFile [in] 分類(ラベル付与)を確認するファイル名(ドライブ文字を含む絶対パス)を指定します。

pbRet [out] 分類(ラベル付与)についてチェックした結果が格納されます。
分類(ラベル付与)されている場合は TRUE、分類(ラベル付与)されていない場合は FALSE が格納されます。

＊ Microsoft 互換形式で分類(ラベル付与のみ)されているファイルのみ TRUE となります。

＊ 分類されていても、Microsoft 互換/FileShell(ラベル)形式で保護されているファイルは FALSE となります。

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または MIP のエラー値が返却されます。

3.3.10 コンテンツ ID 取得 ― IcfsGetContentIdEx()

《概要》

分類/保護ファイルのコンテンツ ID 取得処理をおこないます。

《構文》

```
HRESULT WINAPI IcfsGetContentIdEx(
        LPCWSTR     pszFile,            // [in] 取得するファイルパス
        GUID        *pContentId         // [out] コンテンツ ID
);
```

《引数》

pszFile [in] コンテンツ ID を取得する分類/保護ファイル名(ドライブ名を含む絶対パス)を指定します。

pContentId [out] 分類/保護ファイルのコンテンツ ID が格納されます。

＊ Microsoft 互換形式として保護された PDF ファイルはコンテンツ ID を取得できません。

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.11 権利ポリシー作成 ― IcfsCreateRightsPolicyFile()

《概要》

FileShell 権利ポリシー構造体に指定した内容に従って、ローカル権利ポリシーファイル（xml）を生成します。

《構文》

```
HRESULT WINAPI IcfsCreateRightsPolicyFile(
        ICF_RIGHTS_POLICY    *pRightsPolicy,  // [in] FileShell 権利ポリシー構造体
        LPCWSTR              pszFilePath      // [in] 権利ポリシーファイルの出力ファイルパス
);
```

《引数》

pRightsPolicy [in] FileShell 権利ポリシー構造体（ICF_RIGHTS_POLICY）を指定します。あらかじめ、指定したい権限に従って値を設定しておく必要があります。

参照 FileShell 権利ポリシー構造体の指定方法は「3.3.11.1 FileShell 権利ポリシー構造体 ― ICF_RIGHTS_POLICY」を参照してください。

pszFilePath [in]　ローカル権利ポリシーファイルの出力ファイル名(ドライブ文字を含む絶対パス)を指定します。ここで指定された出力ファイル名でローカル権利ポリシーファイル（xml）が出力されます。

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.11.1 FileShell 権利ポリシー構造体 ― ICF_RIGHTS_POLICY

《概要》

権利ポリシー作成時に指定したい権限に従って、あらかじめ値を設定します。

《定義》

本構造体に関連する情報は IcfRptMaker.h で定義されています。

■ICF_RIGHTS_POLICY ： FileShell 権利ポリシー

型	メンバー名	内容
ICF_NAME_AND_DESC *	pNameAndDesc	権利ポリシー名、説明 * 省略(NULL 指定)できません。
DWORD	dwNameAndDescNum	権利ポリシー名、説明の指定数 * 1 以上を指定してください。
ICF_USER_AND_RIGHT *	pUserAndRight	ユーザー、権限 * 設定しない場合は省略(NULL 指定)可能です。
DWORD	dwUserAndRightNum	ユーザー、権限の指定数 * 0 以上を指定してください。
ICF_CONTENT_EXPIRATION_DATE *	pExpDate	コンテンツの有効期限 * 設定しない場合は省略(NULL 指定)可能です。
BOOL *	pbOwnerFullControl	所有者(作成者)に無期限のフルコントロールの権利を付与するかどうか。 ・TRUE:付与する ・FALSE:付与しない * 省略(NULL 指定)した場合は TRUE となります。
UINT *	puiEULIntervalDay	使用ライセンスの有効期限 * 省略(NULL 指定)した場合は期限なしとなります。
BOOL *	pbCacheInvalid	コンテンツを使用するたびに、新しい使用ライセンスを要求するかどうか。 ・TRUE:要求する ・FALSE:要求しない * 省略(NULL 指定)した場合は FALSE となります。
BOOL *	pbAddonAllowed	ブラウザのアドオンを使用した保護コンテンツの表示を許可する ・TRUE:許可する ・FALSE:許可しない * 省略(NULL 指定)した場合は FALSE となります。
ICF_APP_DATA *	pAppData	アプリケーションの追加情報(拡張ポリシー) * 設定しない場合は省略(NULL 指定)可能です。
DWORD	dwAppDataNum	アプリケーションの追加情報(拡張ポリシー)の指定数 * 0 以上を指定してください。
LPTSTR	pszRefferalURL	権利の要求 URL * 設定しない場合は省略(NULL 指定)可能です。

ICF_REVOCATION *	pRevocation	失効ポリシー * 失効ポリシーは指定可能ですが、使用することはできません。省略(NULL 指定)してください

■ICF_NAME_AND_DESC ： 権利ポリシー名、説明

型	メンバー名	内容
LPTSTR	pszName	権利ポリシー名
LPTSTR	pszDesc	説明
UINT	lcid	言語 ID ・1041： 日本語 ・1033： 英語 ・2052： 中国語(簡体字) ・1028： 中国語(繁体字)

■ICF_USER_AND_RIGHT ： ユーザー、権限

型	メンバー名	内容
LPWSTR	pszUserName	以下を指定。 ユーザー・グループを指定する場合: ユーザー・グループのメールアドレス 例)"samplegrp@sample.com" すべてのユーザーを指定する場合: NULL
LPWSTR	pszUserId	以下を指定。 ユーザー・グループを指定する場合: NULL すべてのユーザーを指定する場合: "ANYONE"
LPWSTR	pszUserIdType	以下を指定。 ユーザー・グループを指定する場合: ICF_RPT_USER_ID_TYPE_UNSPECIFIED すべてのユーザーを指定する場合: ICF_RPT_USER_ID_TYPE_INTERNAL
DWORD	dwRight	指定する権限について、以下の値をビットの論理和で指定。 ICF_RPT_RIGHT_FULLCONTROL ：フルコントロール ICF_RPT_RIGHT_EDIT ：編集 ICF_RPT_RIGHT_VIEW ：表示 ICF_RPT_RIGHT_PRINT ：印刷 ICF_RPT_RIGHT_EXTRACT ：抽出 ICF_RPT_RIGHT_OBJMODEL ：マクロの許可 ICF_RPT_RIGHT_VIEWRIGHTSDATA ：権利の表示

■ICF_CONTENT_EXPIRATION_DATE ： コンテンツの有効期限

型	メンバー名	内容
SYSTEMTIME	stFrom	有効期限の開始日時
SYSTEMTIME	stUntil	有効期限の終了日時

■ICF_APP_DATA ： アプリケーションの追加情報(拡張ポリシー)

型	メンバー名	内容
LPWSTR	pszName	値名
LPWSTR	pszValue	値

3.3.12 ファイルフォーマット取得 ― IcfsGetFileFormat()

《概要》

分類/保護されたファイルのファイルフォーマットを取得します。

《構文》

```
HRESULT IcfsGetFileFormat(
        LPCWSTR     pszFile,        // [in] 分類/保護されたファイル名
        DWORD       *pdwFormat      // [out] ファイルフォーマット
);
```

《引数》

pszFile [in] 保護されたファイル名(ドライブ名を含む絶対パス)を指定します。

**pdwFormat* [out] 保護されたファイル名で指定したファイルのファイルフォーマットが格納されます。格納される値とファイルフォーマットの関係は以下のとおりです。

0: 保護されていない
1: Office IRM/FileShell 形式
2: マルチデバイス形式
＊以下の拡張子については、Microsoft 互換/FileShell(ラベル)形式で保護されたファイルでも、マルチデバイス形式として認識されます。
pfile,ptxt,pxml,pjpg,pjpeg,ppng,ptif,ptiff,pbmp,pgif,pjpe,pjfif,pjt,pjif,pjfi
4: NFP 形式
16: Microsoft 互換/FileShell 形式 (ラベル付与による分類と保護)
32: Microsoft 互換形式 (ラベル付与による分類のみ)

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.3.13 FileShell SDK の終了処理 ― IcfsUninitialize()

《概要》

FileShell SDK の終了処理を実施します。**この処理は FileShell SDK の利用を終了するときに、必ず最後に一度呼び出す必要があります。**

《構文》

HRESULT WINAPI IcfsUninitialize();

＊ FileShell SDK の利用を開始する際には、IcfsInitialize で初期化処理をする必要があります。

参照 IcfsInitialize については、「3.3.1 FileShell SDK の初期化処理 ― IcfsInitialize()」を参照してください。

《引数》

なし

《戻り値》

関数が成功した場合、S_OK が返却されます。失敗した場合、Windows 標準または RMS のエラー値が返却されます。

3.4 エラーコード

FileShell SDK が提供する関数は、エラーコードとして、winerr.h で定義される通常のエラーコード、もしくは下記 URL で定義される AD RMS 独自のエラーコードを返却します。

AD RMS Function Error Codes (2023/12/15 時点)
http://msdn.microsoft.com/en-us/library/bb204613

その他、FileShell 独自のエラーコードとして以下が返却される場合があります。

エラーコード	原因	対処策
0x00040302	既にファイルが保護されているファイルを保護しようとした。	原因欄に記載の内容のとおりです。
0x00040303	既にファイルが保護解除されているファイルを保護解除しようとした。	同上
0x00040305	サイズが 0 バイトのファイルを保護しようとした。	サイズが 0 バイトのファイルは保護できません。
0x80040500	指定された権限で既にファイルが保護されている。	原因欄に記載の内容のとおりです。
0x0004030B	保護または保護解除後のファイル名と同じ名前のファイルが存在します。	マルチデバイス形式での保護・保護解除後、ファイルの拡張子が変更される場合があります。同じ名前のファイルが存在するか確認後、再度保護・保護解除してください。
0x0004030D	権利ポリシーテンプレート情報が見つからない。	保護対象フォルダーの設定に問題がある可能性があります。詳しくは管理者にお問い合わせください。
0x0004030E	原因不明のエラーが発生しました。破損したファイルである可能性があります。	事象ごとに個別に調査する必要があります。
0x0004030F	ネットワークに接続されていないか、サーバーが見つからない。	ネットワークの設定を確認してください。 ネットワークドライブをアンマウントしていないか、最近接続を解除した(未認証状態)ネットワークがないか、確認してください。
0x80040800	Excel 4.0 マクロシート入りファイルを保護しない設定を実施しているにもかかわらず、同ファイルを保護しようとした。	保護したい場合は、保護しない設定を解除してください。
0x80040801	Excel 5.0 モジュールシート入りファイル保護しない設定を実施しているにもかかわらず、同ファイルを保護しようとした。	同上
0x80040802	アドインファイルを保護しない設定を実施しているにもかかわらず、同ファイルを保護しようとした。	同上
0x80040803	ファイル構造と拡張子が一致しない。	拡張子が変更されていないかご確認ください。
0x80040900	RMS Client 2.1 がインストールされていない。	RMS Client 2.1 をインストールしてください。
0x80040901	認証ダイアログでキャンセルした	認証を実施してください。
0x80040950	未変換の NFP 権利ポリシーを使用した。	NFP 権利ポリシーを、権利ポリシーの管理からインポートしてご利用ください。

0x80040953	NFP の緊急保護解除機能で使用する公開鍵の取得に失敗した。	緊急保護解除機能をオフと設定するか、緊急保護解除用の公開鍵を設定してください。
0x80042000	マルチデバイス形式ファイルでない。	マルチデバイス形式ファイルを指定してください。
0x80042001	マルチデバイス形式で保護または保護解除するとき、出力ファイル名が入力ファイル名と同一になる。	ファイル名をご確認ください。
0x80044000	無効な入力エラー。	AIP の引数を見直してください。
0x80044001	バッファーのメモリ不足。	AIP の引数を見直してください。
0x80044002	ファイル IO エラー。	AIP に与えたファイルの状態を確認してください。
0x80044003	ネットワーク エラー。	ネットワークの状態を確認してください。
0x80044004	内部エラー。	SDK の設定や Microsoft Purview コンプライアンス ポータルにてラベルの状態を見直してください。
0x80044005	ファイルに対するアクションが完了できない	Microsoft Purview コンプライアンス ポータルにてラベルの状態を見直してください。
0x80044007	現在のラベルは特権操作（管理者の操作と同等）として割り当てられている	Microsoft Purview コンプライアンス ポータルにてラベルの状態を見直してください。
0x80044008	ユーザーがコンテンツにアクセスできない	Microsoft Purview コンプライアンス ポータルにてラベルの状態を見直してください。
0x80044009	同意が得られなかった。	SDK の設定を見直してください。
0x80044010	コンテンツにアクセスできない。	ファイルに権限があることを確認してください。
0x80044011	認証トークンがない。	SDK の設定や、Azure Portal の設定を見直してください。
0x80044012	サービスが無効になっている。	Azure Portal の設定を見直してください。
0x80044013	プロキシ認証エラー。	ネットワークの状態を確認してください。
0x80044014	テナントポリシーが分類/ラベルに対して構成されていない。	Microsoft Purview コンプライアンス ポータルにてラベルの設定を見直してください。
0x80044015	操作が取り消された。	再試行してください。
0x80044016	アドホック保護を設定する必要がある。	別のラベルを指定してください。
0x80044017	呼び出し元が非推奨の API を呼び出した。	再インストールしてください。
0x80044018	テンプレート ID が RMS サービスによって認識されない。	Microsoft Purview コンプライアンス ポータルにてラベルの設定を見直してください。
0x80044019	ラベル ID が認識されていない。	Microsoft Purview コンプライアンス ポータルにてラベルの設定を見直してください。

0x80044020	ラベルが無効または非アクティブ。	Microsoft Purview コンプライアンスポータルにてラベルの設定を見直してください。
0x80044021	ダブルキー機能が有効になっていない。	Microsoft Purview コンプライアンスポータルにてラベルの設定を見直してください。
0x80044022	ライセンスが登録されていない。	Azure portarl の設定を見直してください。
0x80044501	アドホックのラベルは使用できない。	別のラベルを指定してください。
0x80044502	既に保護されている。	保護を解除してからやり直してください。

3.5 サンプルコード

FileShell SDK を利用したサンプルコードを掲載します。

サンプル 1

以下のサンプルでは、権利ポリシー作成(IcfsCreateRightsPolicyFile)、ファイルの保護(IcfsProtectEx)、保護状態確認(IcfsIsProtectedEx)、コンテンツ ID 取得(IcfsGetContentIdEx)、保護解除(IcfsUnprotectEx)という一連の処理をおこなっています。

```
#include "stdafx.h"
#include <windows.h>
#include "icfsprotector.h"

#pragma comment(lib, "icfsprotector.lib")

int _tmain(int argc, _TCHAR* argv[])
{
   BOOL bRet = FALSE;
   GUID guid;
   HRESULT hr;

   ICF_RIGHTS_POLICY sRightsPolicy = {0};
   ICF_NAME_AND_DESC sNameAndDesc = {0};
   ICF_USER_AND_RIGHT sUserAndRight = {0};
   BOOL bOwnerFullControl = TRUE;
   UINT uiEULIntervalDay = 7;
   BOOL bCacheInvalid = FALSE;
   BOOL bAddonAllowed = FALSE;

   wchar_t plainwbuf[MAX_PATH] = _T("C:¥¥test¥¥plain.txt");
   wchar_t encryptwbuf[MAX_PATH] = _T("C:¥¥test¥¥encrypt.txt");
   wchar_t decryptwbuf[MAX_PATH] = _T("C:¥¥test¥¥decrypt.txt");
   wchar_t templatewbuf[MAX_PATH] = _T("C:¥¥test¥¥RMSTemplate.xml");

   // Create a rights policy template
   sNameAndDesc.lcid = 1041;
   sNameAndDesc.pszName = L"Sample Rights Policy Template";
   sNameAndDesc.pszDesc = L"This is a sample of a rights policy template.";
   sRightsPolicy.pNameAndDesc = &sNameAndDesc;
   sRightsPolicy.dwNameAndDescNum = 1;

   sUserAndRight.pszUserName = L"user@sample.com";
   sUserAndRight.pszUserId = NULL;
   sUserAndRight.pszUserIdType = ICF_RPT_USER_ID_TYPE_UNSPECIFIED;
   sUserAndRight.dwRight = ICF_RPT_RIGHT_FULLCONTROL;
   sRightsPolicy.pUserAndRight = &sUserAndRight;
   sRightsPolicy.dwUserAndRightNum = 1;

   sRightsPolicy.pExpDate = NULL;
   sRightsPolicy.pbOwnerFullControl = &bOwnerFullControl;
   sRightsPolicy.puiEULIntervalDay = &uiEULIntervalDay;
   sRightsPolicy.pbCacheInvalid = &bCacheInvalid;
   sRightsPolicy.pbAddonAllowed = &bAddonAllowed;
   sRightsPolicy.pAppData = NULL;
   sRightsPolicy.dwAppDataNum = 0;
   sRightsPolicy.pszRefferalURL = NULL;
   sRightsPolicy.pRevocation = NULL;

   hr = IcfsInitialize();
```

```
    if (S_OK == hr)
        printf("IcfsInitialize OK!!!¥n");
    else
        printf("IcfsInitialize NG(%08x)!!!¥n", hr);

    hr = IcfsCreateRightsPolicyFile((ICF_RIGHTS_POLICY*)&sRightsPolicy, (LPCWSTR)templatewbuf);
    if (S_OK == hr)
        printf("IcfsCreateRightsPolicyFile OK!!!¥n");
    else
        printf("IcfsCreateRightsPolicyFile NG(%08x)!!!¥n", hr);

    // File Protection
    hr = IcfsProtectEx( (LPCWSTR)plainwbuf, (LPCWSTR)encryptwbuf, (LPCWSTR)templatewbuf);
    if (S_OK == hr)
        printf("IcfsProtectEx OK!!!¥n");
    else
        printf("IcfsProtectEx NG(%08x)!!!¥n", hr);

    //Confirm whether protected file or not
    hr = IcfsIsProtectedEx( (LPCWSTR)encryptwbuf, &bRet);
    if (S_OK == hr) {
        printf("IcfsIsProtectedEx OK!!!¥n");

        if (TRUE == bRet) {
            printf("This file has been already protected.¥n");

            hr = IcfsGetContentIdEx( (LPCWSTR)encryptwbuf, &guid);
            if (S_OK == hr) {
                printf("IcfsGetContentIdEx OK!!!   ContentID :"
                    " {%04x-%02x-%02x-%01x%01x-%01x%01x%01x%01x%01x%01x}¥n",
                    guid.Data1,guid.Data2,guid.Data3,
                    guid.Data4[0],guid.Data4[1],guid.Data4[2],guid.Data4[3],
                    guid.Data4[4],guid.Data4[5],guid.Data4[6],guid.Data4[7]);
            }
            else
                printf("IcfsGetContentIdEx NG(%08x)!!!¥n", hr);

            hr = IcfsUnprotectEx((LPCWSTR)encryptwbuf, (LPCWSTR)decryptwbuf);
            if (S_OK == hr)
                printf("IcfsUnprotectEx OK!!!¥n");
            else
                printf("IcfsUnprotectEx NG(%08x)!!!¥n", hr);

        }
        else
            printf("This file is non-protected. Ready to protect.¥n");
    }
    else
        printf("IcfsIsProtectedEx NG(%08x)!!!¥n", hr);

    hr = IcfsUninitialize();
    if (S_OK == hr)
        printf("IcfsUninitialize OK!!!¥n");
    else
        printf("IcfsUninitialize NG(%08x)!!!¥n", hr);

    return 0;
}
```

サンプル 2

以下のサンプルでは、マルチデバイス形式による ファイルの保護(IcfsProtectMultiDeviceFormat)、保護形式確認(IcfsIsMultiDeviceFormatFile)、保護解除(IcfsUnprotectMultiDeviceFormat)、使用したメモリの解放(IcfsFreeMemory)という一連の処理をおこなっています。

```
#include "stdafx.h"
#include <Windows.h>
#include <tchar.h>
#include <iostream>
#include "icfsprotector.h"
#include "IcfRptMaker.h"

#pragma comment(lib, "icfsprotector.lib")

int main(int argc, _TCHAR* argv[])
{
    BOOL bRet = FALSE;
    HRESULT hr;
    DWORD dwFormat = 0;

    wchar_t plainwbuf[MAX_PATH] = L"C:¥¥test¥¥plain.txt";
    wchar_t multioutputwbuf[MAX_PATH] = L"C:¥¥test";
    wchar_t templatewbuf[MAX_PATH] = L"C:¥¥test¥¥RMSTemplate.xml";

    wchar_t policynamebuf[MAX_PATH] = L"Sample Rights Policy Template";
    wchar_t policydesceiptionbuf[MAX_PATH] = L"This is a sample of a rights policy template.";
    wchar_t usernamebuf[MAX_PATH] = L"user@sample.com";

    wchar_t* pOutputFilePath = NULL;
    wchar_t* pDecryptFilePath = NULL;

    // File Protection (MultiDevice)
   hr = IcfsInitialize();
   if (S_OK == hr)
       printf("IcfsInitialize OK!!!¥n");
   else
       printf("IcfsInitialize NG(%08x)!!!¥n", hr);

    hr = IcfsProtectMultiDeviceFormat((LPCWSTR)plainwbuf, (LPCWSTR)multioutputwbuf,
(LPCWSTR*)&pOutputFilePath, (LPCWSTR)templatewbuf);
    if (S_OK == hr) {
        printf("IcfsProtectMultiDeviceFormat OK!!!¥n");
        printf("IcfsProtectMultiDeviceFormat output file to %ls!!!¥n", pOutputFilePath);

        //Confirm whether protected file or not
        hr = IcfsIsMultiDeviceFormatFile((LPCWSTR)pOutputFilePath, &bRet);
        if (S_OK == hr) {
            printf("IcfsIsMultiDeviceFormatFile OK!!!¥n");

            if (TRUE == bRet) {
                printf("This file has been already protected.¥n");

                hr = IcfsUnprotectMultiDeviceFormat((LPCWSTR)pOutputFilePath, (LPCWSTR)multioutputwbuf,
(LPCWSTR*)&pDecryptFilePath);
                if (S_OK == hr) {
                    printf("IcfsUnprotectMultiDeviceFormat OK!!!¥n");
                    printf("IcfsUnprotectMultiDeviceFormat output file to %ls!!!¥n", pDecryptFilePath);

                    // Memory release
                    IcfsFreeMemory((LPCWSTR)pDecryptFilePath);
```

```
                }
                else
                    printf("IcfsUnprotectMultiDeviceFormat NG(%08x)!!!¥n", hr);

            }
            else
                printf("This file is non-protected. Ready to protect.¥n");
            // Memory release
            IcfsFreeMemory((LPCWSTR)pOutputFilePath);
        }
    }
    else
        printf("IcfsProtectMultiDeviceFormat NG(%08x)!!!¥n", hr);

   hr = IcfsUninitialize();
   if (S_OK == hr)
       printf("IcfsUninitialize OK!!!¥n");
   else
       printf("IcfsUninitialize NG(%08x)!!!¥n", hr);

    return 0;
}
```

サンプル 3

以下のサンプルでは、ファイルフォーマット取得(IcfsGetFileFormat)をおこなっています。

```
#include "stdafx.h"
#include <Windows.h>
#include <iostream>
#include "icfsprotector.h"

#pragma comment(lib, "icfsprotector.lib")

int main(int argc, _TCHAR* argv[])
{
    HRESULT hr;
    DWORD dwFormat = 0;

    wchar_t encryptwbuf[MAX_PATH] = _T("C:¥¥test¥¥encrypt.txt");

    // Format check
   hr = IcfsInitialize();
   if (S_OK == hr)
       printf("IcfsInitialize OK!!!¥n");
   else
       printf("IcfsInitialize NG(%08x)!!!¥n", hr);

    hr = IcfsGetFileFormat((LPCWSTR)encryptwbuf, (DWORD*)&dwFormat);
    if (S_OK == hr) {
        printf("IcfsGetFileFormat OK!!!¥n");

        if (dwFormat == 1)
            printf("This file is IRM Office/FileShell type¥n");
        else if (dwFormat == 2)
            printf("This file is MultiDevice type¥n");
        else if (dwFormat == 4)
            printf("This file is NFP type¥n");
        else if (dwFormat == 16)
            printf("This file is Microsoft compatible/FileShell type¥n");
        else if (dwFormat == 32)
```

```
            printf("This file is Microsoft compatible type¥n");
        else if (dwFormat == 0)
            printf("This file is non-protected¥n");
    }
    else {
        printf("IcfsGetFileFormat NG(%08x)!!!¥n", hr);
    }

   hr = IcfsUninitialize();
   if (S_OK == hr)
        printf("IcfsUninitialize OK!!!¥n");
   else
        printf("IcfsUninitialize NG(%08x)!!!¥n", hr);

    return 0;
}
```

第4章 FileShell SDK 運用環境構築

4.1 運用環境構築の流れ

FileShell SDK の運用環境は以下のような手順で構築します。

No.	項目	内容	参照
1	必要なソフトウェアのインストール	FileShell SDK の利用に必要なソフトウェアをインストールします。	4.2
2	RMS サーバー接続に必要な設定	FileShell SDK が RMS サーバーに接続するために必要な設定をおこないます。 ＊ MIP/Azure RMS サーバーを利用する場合、もしくは NFP 形式のみを利用する場合は不要です。	4.3
3	インストールパッケージの展開	FileShell SDK のインストールパッケージを展開します。	4.5.1
4	インストーラーの作成	利用する保護形式に応じたインストーラーを作成します。	4.4 4.5.2
5	FileShell SDK のインストール	FileShell SDK のインストールを実行します。	4.5.3
6	FileShell SDK の環境設定	FileShell SDK の設定をおこないます。	4.5.4
7	権利ポリシーテンプレートの準備	ファイルの保護に使用するポリシーテンプレートを準備します。 ＊ MIP による分類/保護を利用する場合は不要です。	4.7
8	Azure RMS 権利ポリシーテンプレートの編集	Azure RMS 環境で権利ポリシーテンプレートを編集します。 ＊ MIP による分類/保護を利用する場合、オンプレミス環境で利用する場合、もしくは NFP 形式の保護のみを使用する場合は不要です。	4.8
9	権利ポリシーテンプレートのインポート	FileShell SDK を使ってファイルの保護をおこなう場合に利用する権利ポリシーテンプレートをインポートします。	4.9
10	ラベル ID の取得	MIP による分類/保護に使用するラベル ID を取得します。 ＊ MIP による分類/保護を利用しない場合は不要です。	4.10
11	ラベルへの保護解除権限の付与	FileShell SDK で使用するラベルに対して、保護の解除に必要な権限を付与します。 ＊ MIP による分類/保護を利用しない場合は不要です。	4.11

4.2 必要なソフトウェアのインストール

FileShell SDK の運用環境を構築するには、以下のソフトウェアのインストールが必要です。
サーバーに各ソフトウェアがインストールされていない場合には、手順に従ってインストールを実行してください。

参照 ダウンロード URL については、「1.3 動作環境について」を参照してください。

- Visual Studio 2015、2017、2019、および 2022 用 Visual C++再頒布可能パッケージ
- RMS Client V2.1（NFP 形式のみを使用する場合は不要）

Windows Server 2016 および Windows StorageServer 2016 をご利用の場合は、以下のソフトウェアを動作環境に記載された URL からダウンロードし、インストールしてください

- Microsoft .NET Framework 4.7.2

＊ Windows Server 2019、2022 をご利用の場合は、上記と同じかより新しいバージョンの.NET Framework が OS のデフォルトでインストールされていますので、削除しないでください。

＊ ご利用の OS が日本語環境の場合は、.NET Framework Language Pack をあわせてインストールしてください。

＊ Microsoft 社から提供される最新セキュリティパッチを適用してください。

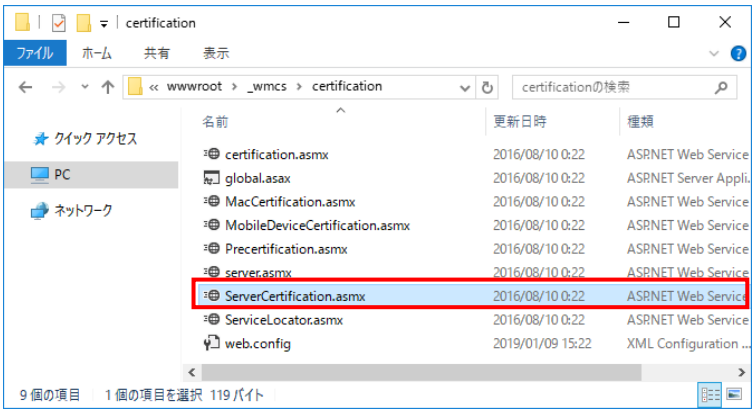
4.3 RMS サーバーへの接続に必要な設定

FileShell SDK では、RMS サーバー接続のために、以下の設定が必要です。

Azure RMS サーバーを利用する場合、もしくは NFP 形式の保護のみを使用する場合は本手順は不要です。

4.3.1 RMS 証明パイプラインに権限を追加する

FileShell SDK から RMS サーバーに要求をするためには、RMS サーバーの RMS 証明パイプラインに FileShell SDK を使用するアカウント、および、AD RMS Service Group を追加する必要があります。



RMS サーバーにて、ServerCertification.asmx ファイル※1 のアクセス許可に、FileShell SDK を実行するアカウント※2、および、AD RMS Service Group※3 を、”読み取りと実行”の権限で追加します。

＊ (※1) ServerCertification.asmx の既定のファイルパスは以下です。

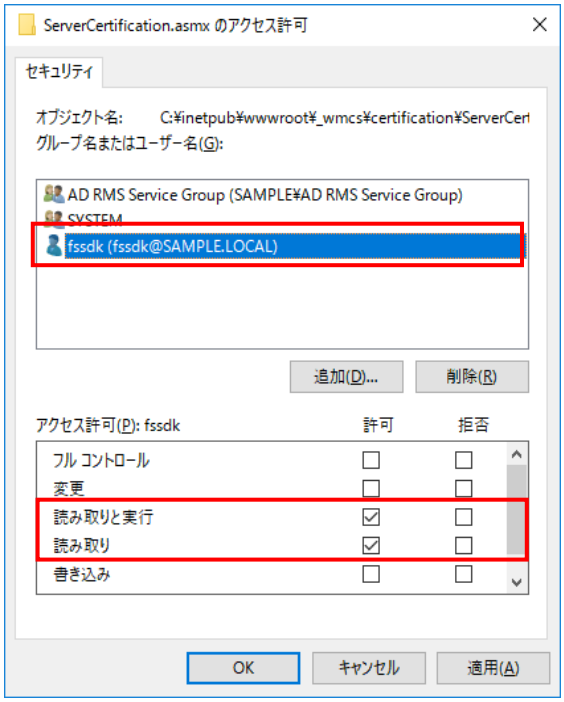
%systemdrive%¥Inetpub¥wwwroot¥_wmcs¥Certification¥ServerCertification.asmx

certification.asmx という別のファイルもあるため、間違えないようにご注意ください。

＊ (※2) System アカウント、Network Service アカウントで実行している場合は、FileShell SDK を実行するマシンのコンピューター アカウント、または、FileShell SDK を実行するマシンのコンピューター アカウントが所属するセキュリティグループを追加してください。

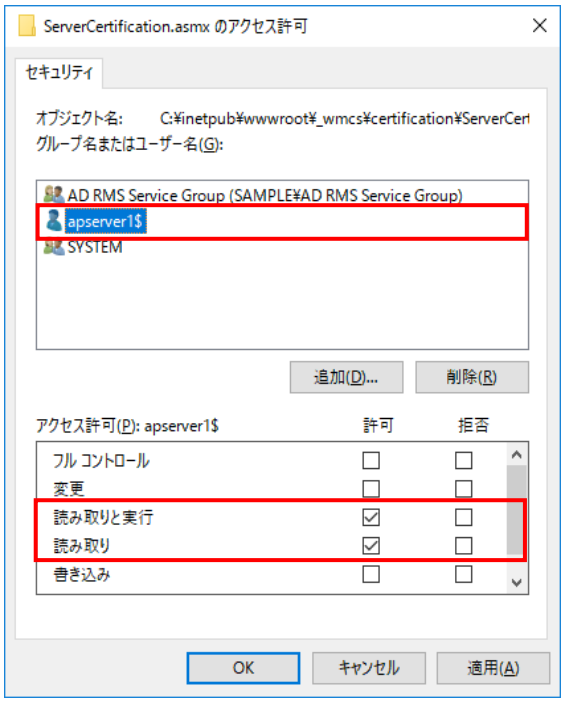
＊ (※3) AD RMS Service Group は AD RMS サーバーのインストール時に作成されるセキュリティグループです。

例） SDK を動かすアプリケーションが、ユーザーfssdk@SAMPLE.LOCAL で動作するとき



例） SDK を動かすアプリケーションが、SYSTEM アカウントで動作するとき

（動作するマシン名が apserver1）



4.3.2 信頼された発行ドメインの設定

RMS の構成である Trusted Publishing Domain(TPD:信頼された発行ドメイン)の環境で利用するためには、以下のレジストリ設定をおこないます。

■ TPD:信頼された発行ドメイン

・下記レジストリを設定してください。
［HKLM¥SOFTWARE¥Microsoft¥MSIPC¥Servicelocation¥LicensingRedirection］

・名前 :保護ファイルを生成した他ドメインの RMS サーバーのライセンスサーバーURL
例）http://domainA/_wmcs/licensing

・種類 :REG_SZ

・データ :保護ファイルを利用する自ドメインの RMS サーバーのライセンスサーバーURL
例）http://domainB/_wmcs/licensing

＊ Trusted User Domain(TUD:信頼されたユーザードメイン)の設定は必要ありません。

＊ オンプレミスの AD RMS サーバーを、Azure RMS サーバーにリダイレクトするときなど、 移行元となるオンプレミスの AD RMS サーバーとしてイントラネットクラスターURL とエクストラネットクラスターURL の両方が設定されており、かつその URL が異なる場合には、両方の URL をリダイレクト元とするライセンスサーバーURL のリダイレクト設定が必要です。

4.4　インストーラー作成時に設定する情報の取得

FileShell SDK のインストーラー作成で必要になる情報を事前に取得します。

4.4.1　MIP/Azure RMS を利用する場合に必要な情報の設定・取得

FileShell SDK で、MIP ラベルでのファイルの分類/保護（Microsoft 互換/FileShell 形式での出力）、もしくは Azure RMS 上の権利ポリシーテンプレートでの保護を利用するには、Azure Portal でアプリケーションの登録・設定をおこない、FileShell 付属の「Azure RMS 接続情報取得ツール」で情報を取得する必要があります。

＊ V6.1 未満の FileShell SDK で、Azure RMS を利用している環境からアップデートする場合も、本項の手順を実施してください。

＊ NFP 形式の保護のみを利用する場合は、本項の手順は不要です。

4.4.1.1　Azure Portal でのアプリケーションの登録

Operation

1. Microsoft Azure Portal (https://portal.azure.com) にアクセスし、テナントのグローバル管理者の資格情報を持つユーザーでログインします。

2. Azure Portal ホームの「Azure Active Directory の管理」の[表示]をクリックします。

3. Azure Active Directory の管理で、メニューの[追加]から[アプリの登録]を選択します。

4. アプリケーションの登録で以下を入力/選択します。

・ 名前:
任意の名前を入力します（本書では"FileShell SDK"と入力することとします）。

・ サポートされているアカウントの種類：
［この組織のディレクトリのみに含まれるアカウント(″テナント名″のみ-シングルテナント)］を選択します。

・ リダイレクト URI：
パブリック クライアント/ネイティブ(モバイルとデスクトップ)を選択し、入力欄には″fileshell://authorize″と入力します。
＊ 設定したリダイレクト URI は、FileShell SDK のインストーラー作成時に、プロパティ″ADAL_URI″に設定する値として使用します。

すべての入力/選択が完了後、[登録]をクリックします。

5. アプリケーションが登録されます。表示されるアプリケーションの情報のうち、以下の情報は、FileShell SDK のインストーラー作成時に使用します。

・アプリケーション(クライアント)ID ：プロパティ″ADAL_ID″に設定する値として使用します。

・ディレクトリ(テナント)ID ：プロパティ″ADAL_TENANT_ID″に設定する値として使用します。

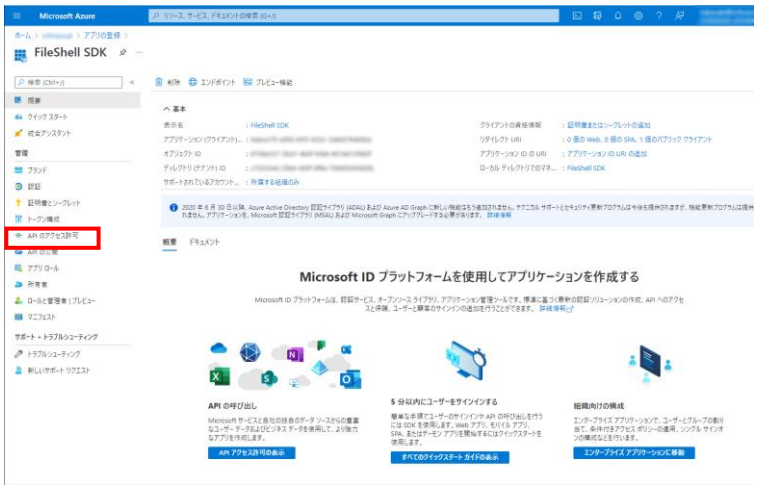
＊ [アプリケーション(クライアント)ID]は、「4.11 ラベルへの保護解除権限の付与」でも使用します。

以上で、アプリケーションの登録は終了です。引き続き、「4.4.1.2 API アクセス許可の追加」を実施してください。

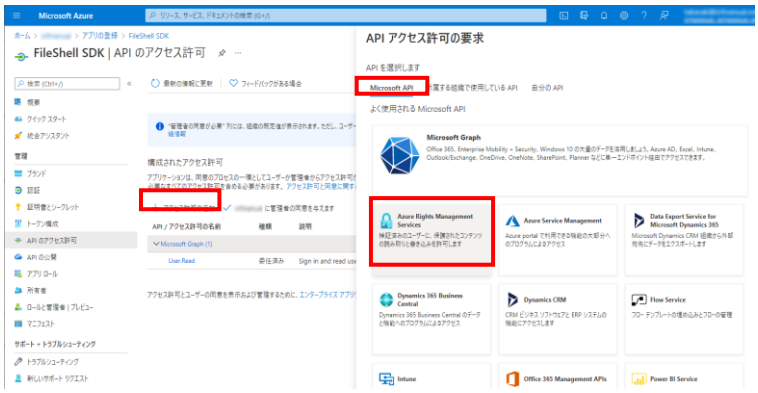
4.4.1.2 API アクセス許可の追加

Operation

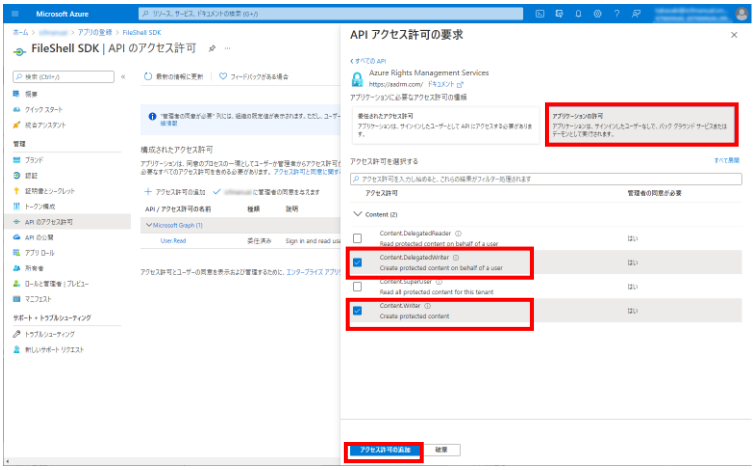
1. 登録したアプリケーションの画面の左メニューの[管理]から[API のアクセス許可]をクリックします。



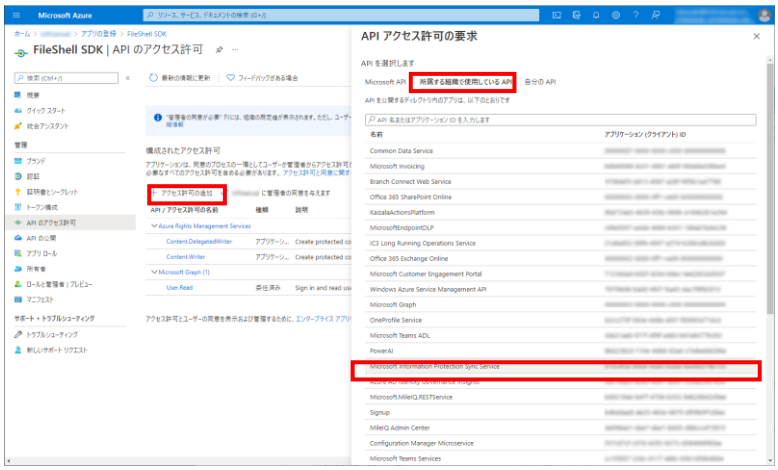
2. API アクセスの許可の要求で、[アクセス許可の追加]をクリックすると表示される「API アクセス許可の要求」で、[Microsoft API]の[Azure Rights Management Service]を選択します。



3. API のアクセス許可要求で[アプリケーションの許可]を選択し、[Content.DelegatedWriter] と、[Content.Writer]にチェックを入れて、[アクセス許可の追加]をクリックします。

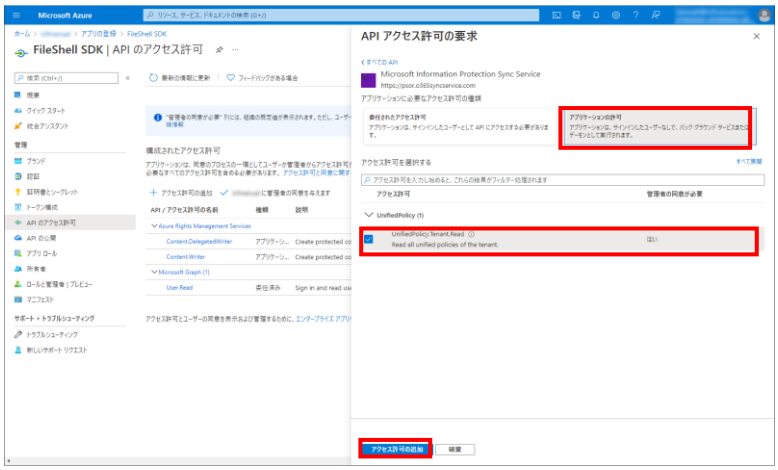


4. 手順 **3** で追加したアクセス許可が追加されたことを確認したのち、もう一度[アクセス許可の追加]をクリックし、API アクセス許可の[所属している組織で使用している API]を選択し、表示される一覧から、[Microsoft Information Protection Sync Service]を選択します。

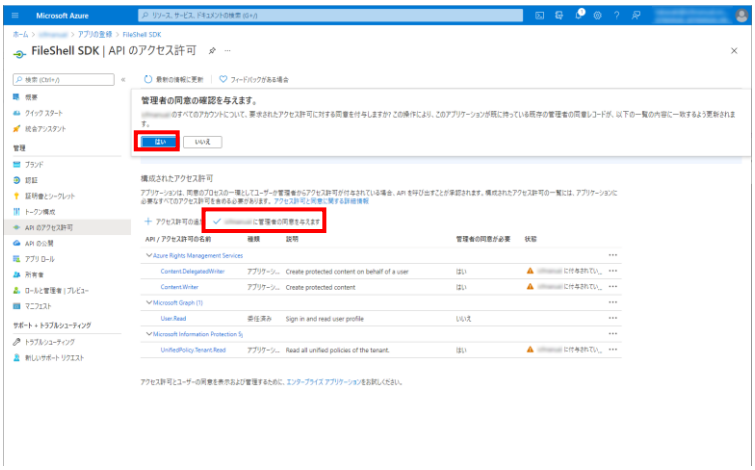


＊ 一覧に[Microsoft Information Protection Sync Service]が表示されていない場合は、一覧下部の[さらに読み込む]をクリックしてください。

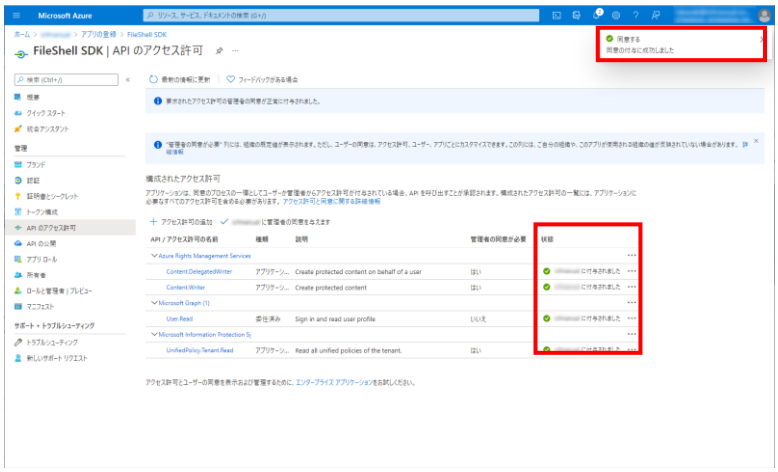
5. API のアクセス許可要求で[アプリケーションの許可]を選択し、[UnifiedPolicy.Tenant.Read]にチェックを入れて、[アクセス許可の追加]をクリックします



6. 手順 **5** で追加したアクセス許可が追加されたことを確認したのち、[(テナント名)の管理者に同意を与えます]をクリックすると、「管理者の同意の確認を与えます。」のメッセージが表示されますので[はい]を選択します。



7. 同意の付与に成功するとメッセージが表示され、構成されたアクセス許可の状態が更新されます。

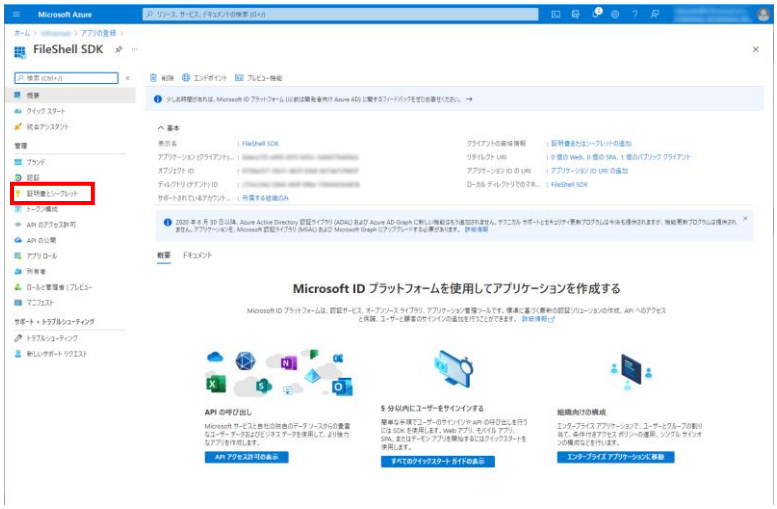


以上で、アプリケーションへの API のアクセス許可の追加は終了です。引き続き「4.4.1.3 クライアントシークレットの作成」を実施してください。

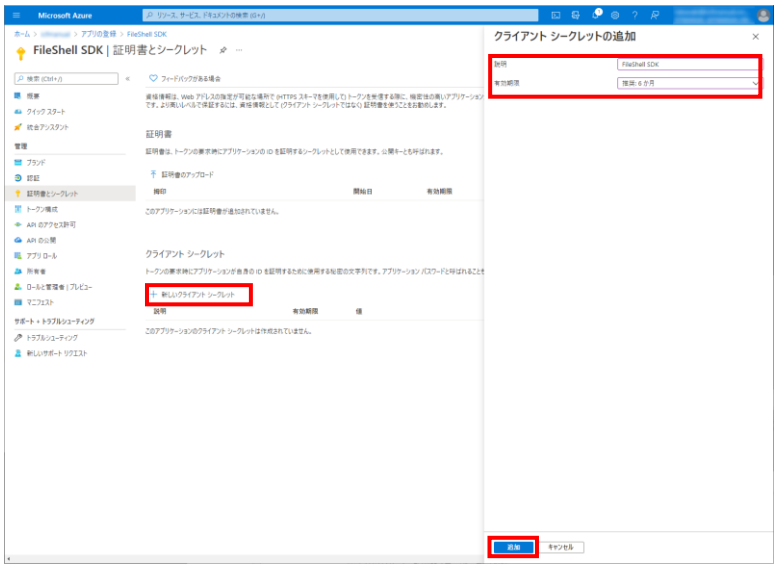
4.4.1.3 クライアントシークレットの作成

Operation

1. 登録したアプリケーションの画面で、左側のメニューの[管理]から[証明書とシークレット]をクリックします。



2. クライアントシークレットの[新しいクライアントシークレット]をクリックすると表示されるクライアントシークレットの追加で、[説明]と[有効期限]を設定し、[追加]をクリックします。



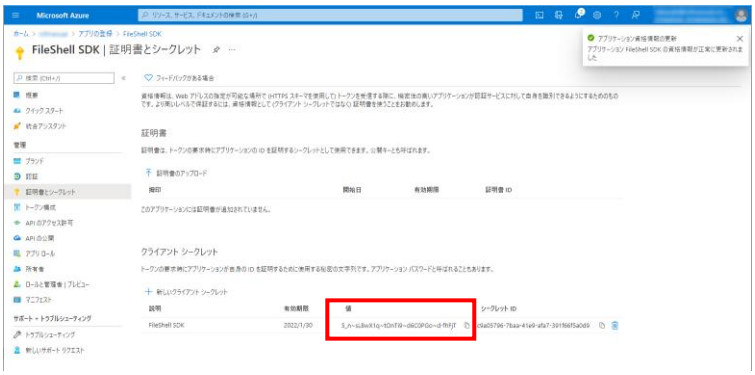
＊ クライアントシークレットの有効期限は最大 24 か月です。運用年数、セキュリティリスクなどを考慮の上、適切な期限を設定し、運用中に期限が切れることのないよう管理をおこなってください。有効期限が切れた場合は、本項に記載の手順でクライアントシークレットを再作成してください。

＊ クライアントシークレットが漏えいすると、Azure Active Diretory からユーザー・グループ・AU などの情報を第三者に取得される可能性があります。クライアントシークレットの管理には十分ご注意ください(漏えいの疑いがある場合は、すぐに Azure Portal でアプリの設定を無効化するなど、処置をおこなってください)。

3. アプリケーションの資格情報が正常に更新された旨のメッセージが表示され、クライアントシークレットが追加されますので、表示されている値を控えます。

＊ ここで表示されるクライアントシークレットの値は、あとから参照することができませんので、**表示されたタイミングで必ず控えるようにしてください。**

＊ 値が表示されている右側の アイコンをクリックすると、値をクリップボードにコピーできます。



以上で、クライアントシークレットの作成は終了です。

4.4.1.4 Azure RMS の接続情報の取得

Operation

1. Windows PowerShell から、本操作に必要となるモジュールをインストールします。
Windows の [スタートメニュー]からプログラムの一覧を表示し、[Windows PowerShell]の右クリックメニューから、「管理者として実行」を選択します。

2. 以下のコマンドを実行し、TLS1.2 を有効にします。

> [Net.ServicePointManager]::SecurityProtocol = [Net.ServicePointManager]::SecurityProtocol -bOR [Net.SecurityProtocolType]::Tls12

＊ お使いの環境が Windows Server 2019 以降の場合は、本コマンドの実行は不要です。

3. 以下のコマンドを実行し、次のモジュールをインストールします。

・ AIPService PowerShell モジュール

> Install-Module -Name AIPService

＊ お使いの環境に Azure Rights Management Administration Tool (AADRM)がインストールされている場合は、以下のコマンドを実行し、AADRM をアンインストールしてから AIPSevice PowerShell モジュールをインストールしてください。

> Uninstall-Module -Name AADRM

＊ パッケージマネージャー「nuget」のインストール確認メッセージが表示された場合は、「Y」を入力して続行してください。

＊ モジュールのインストール時に、「信頼されていないリポジトリからモジュールをインストールしようとしています。..」と表示された場合は、「Y」を入力して続行してください。

4. 以下のコマンドを実行し、モジュール一覧を取得します。

> Get-InstalledModule

5. 一覧の「Name」に手順 **3** でインストールしたモジュールが表示されていることを確認します。

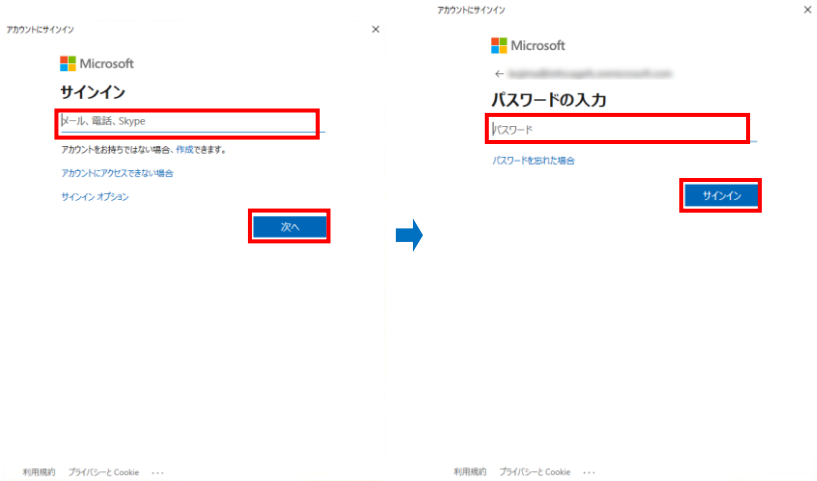
```
PS D:¥                                  > Get-InstalledModule

Version    Name                                Repository         Description
-------    ----                                ----------         -----------
1.0.0.     AIPService                          PSGallery          PowerShell m
```

6. 以下のコマンドを実行します。

> Connect-AIPService

7. 表示されたサインイン画面で Azure のユーザーID とパスワードを入力し、サインインします。

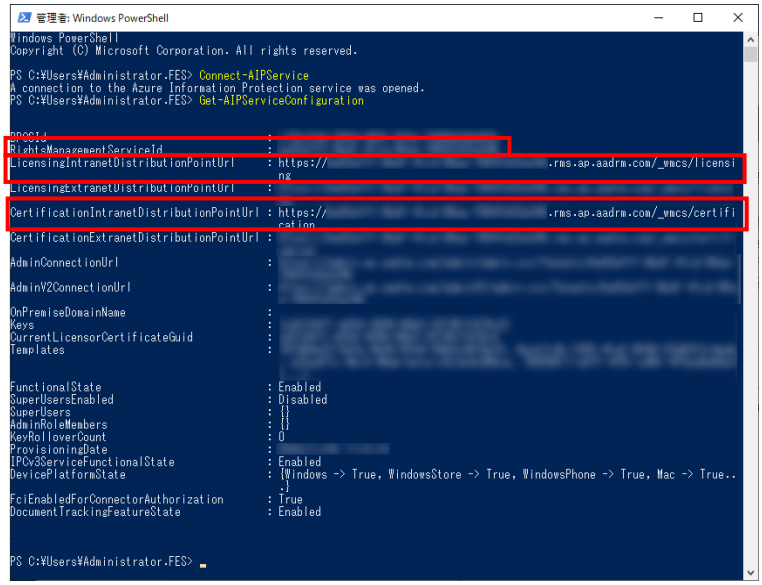


＊ グローバル管理者の資格情報を持つユーザーで接続してください。

8. 以下のコマンドを実行します。

> Get-AIPServiceConfiguration

9. 表示された「RightManagementServiceId」、「LicensingIntranetDistributionPointUrl」(Azure RMS のライセンスサーバーURL)、および「CertificationIntranetDistributionPointUrl」(Azure RMS の認証サーバーURL)を確認します。



＊ 本手順で取得した Azure RMS のライセンスサーバーURL、および認証サーバーURL は、「5.3 環境設定ツールの操作」-「5.3.1 RMS 認証情報設定」で使用します。

4.4.2 NFP の緊急保護解除用の公開鍵の取得

＊NFP 形式の保護を利用しない、または NFP の緊急保護解除機能を使用しない場合は、本項の手順は不要です。

NFP の緊急保護解除機能は、NFP 形式のファイルを保護した権利ポリシーを紛失した場合などに、緊急措置としてファイルの保護を強制的に解除するための機能です。FileShell SDK で NFP 形式の保護を利用する際は、NFP の緊急保護解除機能を有効にして使用することを推奨します。

FileShell SDK で NFP の緊急保護解除機能を有効にするには、インストール設定ファイルにて、緊急保護解除機能を有効にし、公開鍵を設定する必要があります。使用する権利ポリシーに応じて、以下の手順で公開鍵を事前に取得し、次項「4.5.2 インストーラーの作成」で、インストール設定ファイルに設定してください。

＊ 緊急保護解除機能が有効でない環境で保護された NFP 形式のファイルは、保護した権利ポリシーを紛失した場合、ファイルの保護を解除することができません。万一の場合に備え、本機能を有効にして利用することを推奨します。

＊ 緊急保護解除機能による保護解除については『NEC File Protection Edition 管理者ガイド』の「NFP の緊急保護解除機能による保護解除について」を参照してください。

● FileShell サーバーで作成した NFP 権利ポリシーを使用する場合

Operation

1. FileShell サーバーに全体管理者もしくは組織管理者の ID でログインします。
2. [組織管理者用]メニューから[組織設定]をクリックします。

＊ 全体管理者の ID でログインした場合は[組織管理]-[組織一覧]から設定したい組織を選択することで、[組織管理者用]メニューを表示できます。

3. [組織情報設定]画面の[権利ポリシ利用設定]-[公開鍵]欄に表示されている緊急保護解除用の公開鍵を取得します。

＊ 公開鍵が表示されていない場合は、「鍵ペアを作成」ボタンで公開鍵を作成してください。

以上で、NFP の緊急保護解除用公開鍵の取得は終了です

● FileShell クライアントで作成した NFP 権利ポリシーを使用する場合

Operation

1. NFP 権利ポリシーを作成した FileShell クライアントがインストールされている環境で、コマンドプロンプトでカレントを FileShell クライアントのインストール先に移動し、以下のコマンドを実行します。

IcfClientSetting.exe /CreatePolicy

＊ FileShell クライアントは、デフォルトでは以下にインストールされています。

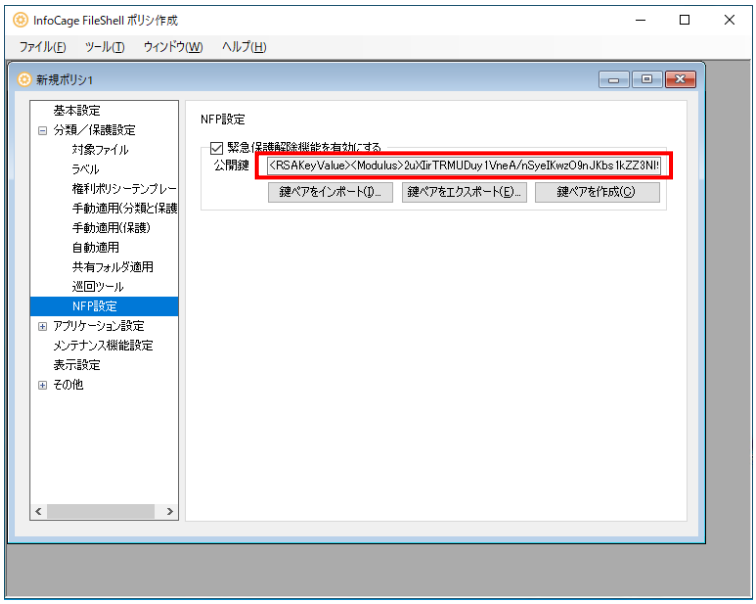
〈システムドライブ〉¥Program Files¥NEC¥InfoCageFileShell¥

2. FileShell ポリシー作成画面が表示されます。サブのメニューツリーから[保護設定]-[NFP 設定]を選択し、[公開鍵]欄に表示されている緊急保護解除用の公開鍵を取得します。

[公開鍵]欄に表示されている文字列をコピーしてご使用ください。

＊ 公開鍵が表示されていない場合は、[鍵ペアを作成]ボタンで公開鍵を作成してください。

＊ 緊急保護解除用の公開鍵は、『NEC File Protection Edition 管理者ガイド』の「緊急保護解除用の公開鍵のエクスポート」に記載の手順で、エクスポートして保存し、厳重に管理してください。



以上で、NFP の緊急保護解除用の公開鍵の取得は終了です。

4.5 FileShell SDK のインストール

FileShell SDK を、運用環境にインストールする手順について説明します

4.5.1 インストールパッケージの展開

運用環境にインストールする FileShell SDK のインストーラーを作成するために必要なパッケージは、以下の手順で展開してください。

1. メディアキット CD-ROM から、FileShell SDK インストーラ格納フォルダー(Setup)をインストールするマシンの任意のフォルダーにコピーしてください。本書では、「D:¥icfadmin¥icfsdk」 にコピーしたと仮定します。

■ メディアキット CD-ROM 内フォルダー構成

フォルダー名	説明
¥Tools	
└Develop	
└ServerSDK	
├Include	C/C++開発用ヘッダファイル格納フォルダー
├Lib	C/C++開発用インポートライブラリ格納フォルダー
│ └x64	64bit 用インポートライブラリ
└Setup	FileShell SDK インストーラ格納フォルダー(本フォルダーをコピーします)
└x64	64bit 用インストーラ格納フォルダー

■ 配置イメージ

[D:¥icfadmin¥icfsdk]フォルダー配下に、[Setup]フォルダー配下一式を配置します。

D:¥icfadmin¥icfsdk
└ ¥Setup

2. メディアキット CD-ROM から、インストーラ作成支援ツールのインストールの格納フォルダー(¥Tools¥SetupConfig) を PC の任意のフォルダーにコピーしてください。本書では、「D:¥icfadmin¥icfsdk」にコピーしたと仮定します。

■ メディアキット CD-ROM 内フォルダー構成

フォルダー名	説明
¥Tools	
└¥SetupConfig	インストーラ作成支援ツール格納フォルダー(本フォルダーをコピーします)

■ 配置イメージ

[D:¥icfadmin¥icfsdk]フォルダー配下に、[SetupConfig]フォルダー配下一式を配置します。

D:¥icfadmin¥icfsdk
├ ¥Setup (手順 *1* で配置したフォルダー)
└ ¥SetupConfig

以上で、インストールパッケージの展開は、終了です。

4.5.2 インストーラーの作成

利用する保護形式に応じたインストーラーを作成します。

インストーラーの作成は、インストーラ作成支援ツールでおこないます。

＊ インストーラーは必ずインストーラ作成支援ツールを使用して作成してください。テキストエディタ等によるインストーラー設定ファイルの編集はおこなえません。

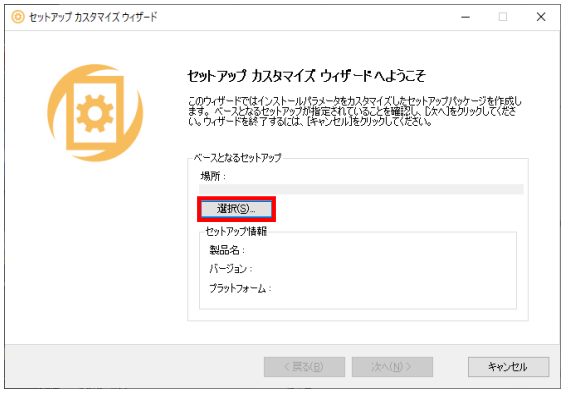
＊ インストーラ作成支援ツールについての詳細は、『インストールガイド』の「インストーラ作成支援ツール」を参照してください。

＊ 本項の手順では、「4.5.1 インストールパッケージの展開」に記載の配置イメージどおりにインストールパッケージを展開したものとして説明しています。インストールパッケージの配置フォルダーを変更している場合は、変更したフォルダーに読み替えて手順をすすめてください。

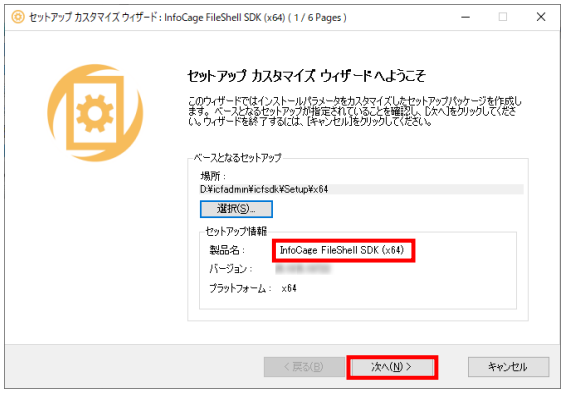
Operation

1. 「D:¥icfadmin¥icfsdk¥SetupConfig¥setupconfig.exe」を実行します。

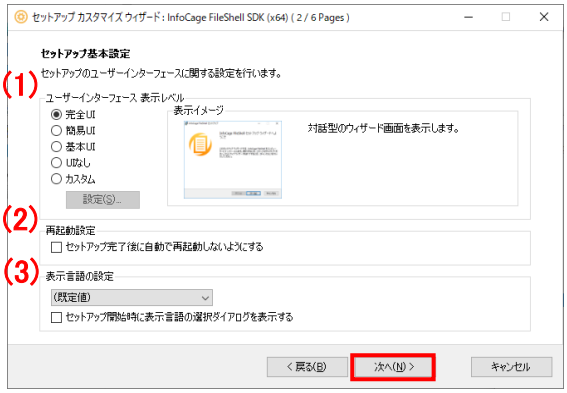
2. インストーラ作成支援ツールが起動し、「セットアップ カスタマイズ ウィザードへようこそ」画面が表示されます。［ベースとなるセットアップの情報］の［選択］ボタンをクリックし、「D:¥icfadmin¥icfsdk¥Setup¥x64¥Setup.exe」を指定します。



3. [セットアップ情報]の[製品名]が、"InfoCage FileShell SDK (x64)"になっていることを確認し、[次へ]をクリックします。

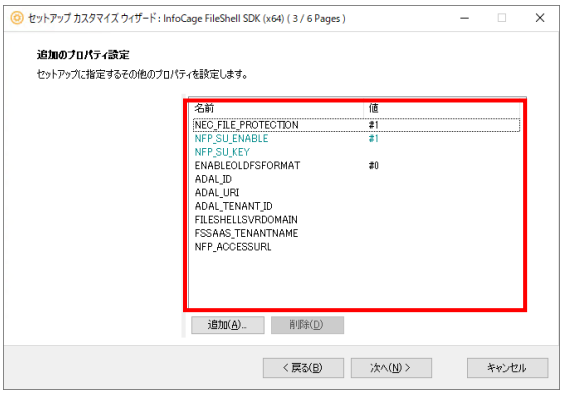


4. 「セットアップ基本設定」画面が表示されます。
以下の項目を必要に応じて設定し、[次へ]をクリックします。



項目		内容
(1)	ユーザーインタフェース表示レベル	UI の表示レベルを選択します。 サイレントインストールをおこなう場合は、[UI なし]を選択します。 ＊ その他の表示レベルについては、[表示イメージ]内の説明をご参照ください。
(2)	再起動設定	FileShell SDK のセットアップでは、自動で再起動はおこなわれないため、設定の必要はありません(チェックを入れなくても、OS の再起動はおこなわれません)。
(3)	表示言語の設定	セットアップ開始時に表示される言語をリストから選択します。 セットアップ開始時に表示言語の選択ダイアログを表示する場合は、チェックボックスにチェックを入れます。

5. 追加のプロパティ設定画面が表示されますので、利用する環境に応じた設定をおこないます。



プロパティ名をダブルクリックすると、プロパティの編集画面が表示されます。利用する形態に応じて以下の一覧を参考に、プロパティの[値のデータ]の編集をおこなってください。

＊ 画面上に表示されていないプロパティは[追加]ボタンで追加することができます。

＊ FileShell SDK で MIP による分類／保護を利用する場合は、以下のプロパティの設定が必要です。

- ADAL_ID
  「4.4.1.1 Azure Portal でのアプリケーションの登録」の手順 **5** で確認した「アプリケーション(クライアント)ID」を指定してください。
- ADAL_URI
  「4.4.1.1 Azure Portal でのアプリケーションの登録」の手順 **4** で設定した「リダイレクト URI」を指定してください。
- ADAL_TENANT_ID
  「4.4.1.1 Azure Portal でのアプリケーションの登録」の手順 **5** で確認した「ディレクトリ(テナント) ID」を指定してください。

＊ 以下に該当する場合は、プロパティ"MSIPC_MSAL_ENABLE"を、設定値"#0"(MSIPC で MSAL での認証を使用しない)で追加してください。

- オンプレミスの AD RMS のみを利用する場合
- NFP のみを利用する場合

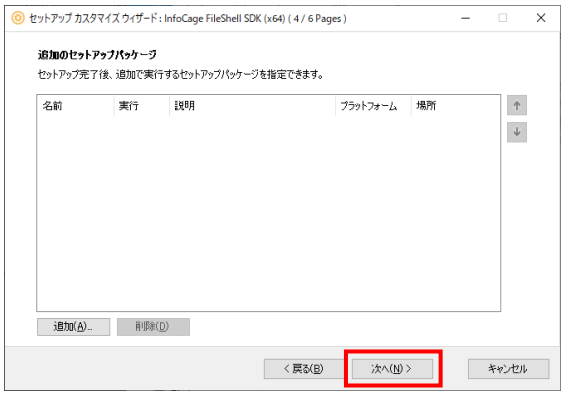
＊ NFP の緊急保護解除に関する設定については、「4.5.2.1 NFP の緊急保護解除に関する設定例」に、利用形態ごとの設定例を記載しておりますので、参考にしてください。

プロパティ名	内容	設定
NEC_FILE_PROTECTION	NFP 機能の有効／無効を設定します。	#0：無効 #1：有効(既定値)
NFP_SU_ENABLE	NFP の緊急保護解除機能を有効／無効を設定します。 ＊ 緊急保護解除機能が有効でない環境で保護された NFP 形式のファイルは、保護した権利ポリシーを紛失した場合、ファイルの保護を解除することができません。NFP 機能を利用する場合は、万一の場合に備え、本機能を有効にして利用することを推奨します。 ＊ 有効にする場合は、NFP_SU_KEY に、公開鍵文字列の設定が必要です。	#0：無効 #1：有効(既定値)
NFP_SU_KEY	NFP の緊急保護解除機能で使用する公開鍵文字列を指定します。	NFP の緊急保護解除機能を利用する場合：「4.4.2 NFP の緊急保護解除用の公開鍵の取得」で取得した公開鍵を、"文字列"形式で指定します。 NFP の緊急保護解除機能を利用しない場合：空欄とします。
ENABLEOLDFSFORMAT	V5.0 以前のクライアントと互換性のある FileShell 形式の保護を有効にします。(FileShell 形式有効設定) ＊ 本設定を有効に設定した場合、MIP による分類／保護をおこなってもラベルは付与されません。	#0：無効(既定値) #1：有効
MSIPC_MSAL_ENABLE	MSIPC で MSAL での認証を使用する設定です。 本設定は、Azure RMS への認証を MSIPC で利用する場合に有効にします。 ＊ 以下に該当する場合は、本プロパティの設定を" #0"(MSIPC で MSAL での認証を使用しない)に設定してください。 ・オンプレミスの AD RMS のみを利用する場合 ・NFP のみを利用する場合	#0 ： MSIPC で MSAL での認証を使用しない #1 ： MSIPC で MSAL での認証を使用する(既定値)
ADAL_ID	MIP/Azure RMS を利用する場合に、「4.4.1.1 Azure Portal でのアプリケーションの登録」の手順 5 で確認した「アプリケーション(クライアント)ID」を指定します。	"文字列"形式で指定します。

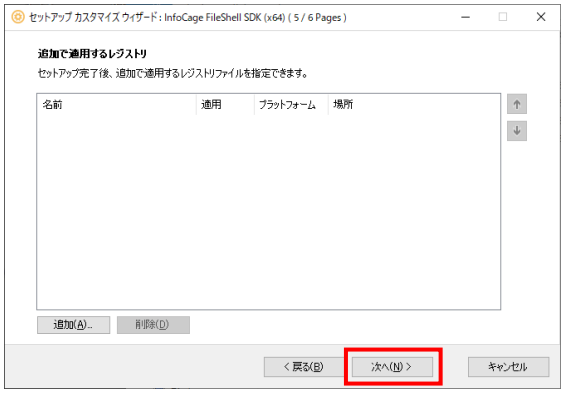
ADAL_URI	MIP/Azure RMS を利用する場合に、「4.4.1.1 Azure Portal でのアプリケーションの登録」の手順 *4* で設定した「リダイレクト URI」を指定します。	"文字列"形式で指定します。
ADAL_TENANT_ID	MIP/Azure RMS を利用する場合に、「4.4.1.1 Azure Portal でのアプリケーションの登録」の手順 *5* で確認した「ディレクトリ(テナント)ID」を指定します。	"文字列"形式で指定します。
FILESHELLSVRDOMAIN	ID 配布方式でポリシーの配布先を管理する組織を利用する場合に、その組織の組織 ID として設定しているドメイン名を指定します。	"文字列"形式で指定します。 ＊ 簡易配布でポリシーの配布先を管理する組織を利用する場合は、本プロパティを設定しないでください。
FSSAAS_TENANTNAME	簡易配布でポリシーの配布先を管理する組織を利用する場合に、その組織の組織 ID を指定します。	"文字列"形式で指定します。 ＊ ID 配布でポリシーの配布先を管理する組織を利用する場合は、本プロパティを設定しないでください。
NFP_ACCESSURL	サーバー認証版の NFP 権利ポリシーで使用する FileShell サーバーの URL を指定します。 ＊ 本設定は、サーバー認証版の NFP 権利ポリシーテンプレートを使用する場合のみ必要です。 ＊ サーバー認証版の NFP 権利ポリシーテンプレートを使用する場合、本設定がおこなわれていないと保護することができません。	"文字列"形式で指定します。 ＊ http:// 、もしくは https://で始まる URL を指定ください。 ＊ スペース(全角、半角)を含む URL は指定できません。

設定完了後、[次へ]をクリックします。

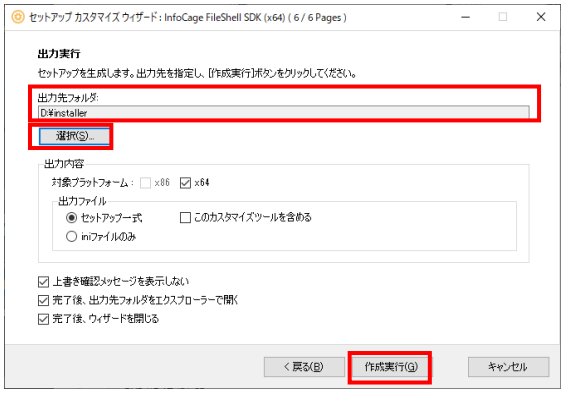
6. [追加のセットアップパッケージ]画面が表示されますので、[次へ]をクリックします。



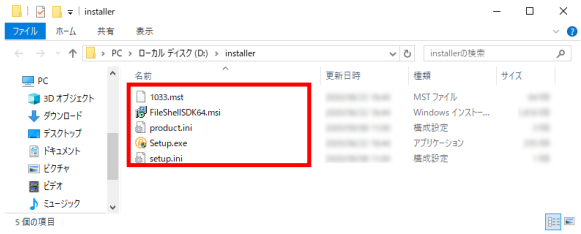
7. [追加で適用するレジストリ]画面が表示されますので、[次へ]をクリックします。



8. 出力実行画面で[選択]ボタンをクリックし、インストールパッケージの出力先フォルダーを指定して[作成実行]ボタンをクリックします。



9. 出力先に指定したフォルダーに、FileShell SDK のインストールパッケージが作成されます。



以上で、FileShell SDK のインストーラーの作成は終了です

4.5.2.1 NFP の緊急保護解除に関する設定例

NFP 形式の保護を利用しない場合の設定

プロパティ名	内容	値の設定
NEC_FILE_PROTECTION	NFP を無効に設定します。	#0 ＊[値のデータ]には#を入力しないでください。
NFP_SU_ENABLE	緊急保護解除機能を無効に設定します。	#0 ＊[値のデータ]には#を入力しないでください。
NFP_SU_KEY	空欄のままとします。	

NFP 形式の保護を利用する場合の設定（緊急保護解除機能有効）

プロパティ名	内容	値の設定
NEC_FILE_PROTECTION	NFP を有効に設定します。	#1 ＊［値のデータ］には#を入力しないでください。
NFP_SU_ENABLE	緊急保護解除機能を有効に設定します。	#1 ＊［値のデータ］には#を入力しないでください。
NFP_SU_KEY	「4.4.2 NFP の緊急保護解除用の公開鍵の取得」で取得した NFP の緊急保護解除用の公開鍵を設定します。	欄外参照

NFP_SU_KEY の設定例

＊ サンプルです。値には実際に取得したものを設定してください

<RSAKeyValue><Modulus>rTaBBbKqsUIncMAKwhmGBiK/c8/5yfpZ0F4snrllMRZVNQf/Os
NyiGtQoMUr7WFHGgibPJ9u9K1+XpG+NcllORWJ2/bQiemdA0fKEtrjEMH0nu7MsqRWtU0
6gopb5BTCOUuGOodn1GyYMHgLAbTwMg3MzgHpfns5PyqxXJIZrEZBfyAlP5jyVuEJu/IxF
KW18vTCgTtb+x6zR2dPtQEZAua2koUwf4imZKv0PEOIfxAF3rW6I7Wm4c5cHt3oiGprFAS
DMPJPz1GS/8ulZ5fF2iukUYy16FYbObD1PeWcnFos9QkF4eqhlMWjW8Xfa99HtBbZVKuN
CCKBngQ4o7fTWw==</Modulus><Exponent>AQAB</Exponent></RSAKeyValue>

NFP 形式の保護を利用する場合の設定（緊急保護解除機能無効）

＊ NFP の緊急保護解除機能を無効にしてのご利用は推奨しません

プロパティ名	内容	値の設定
NEC_FILE_PROTECTION	NFP を有効に設定します。	#1 ＊［値のデータ］には#を入力しないでください。
NFP_SU_ENABLE	緊急保護解除機能を無効に設定します。	#0 ＊［値のデータ］には#を入力しないでください。
NFP_SU_KEY	空欄のままとします。	

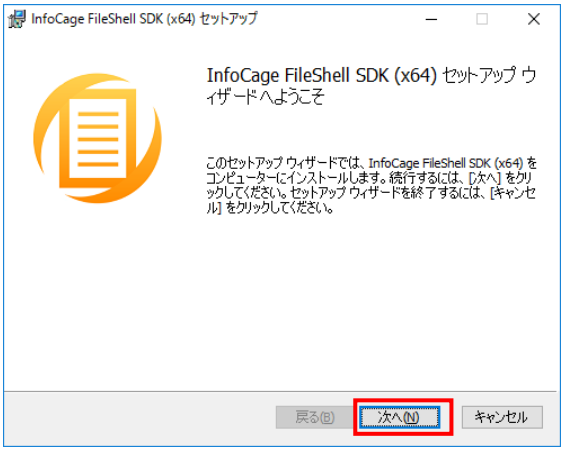
4.5.3 インストール

FileShell SDK をインストールする手順を説明します。

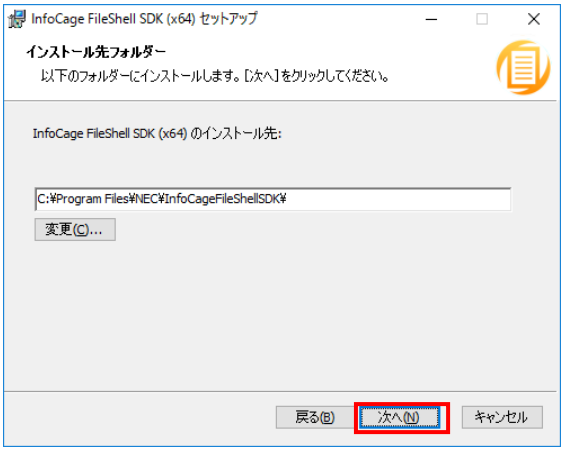
Operation

1. 「4.5.2 インストーラーの作成」で作成したインストールパッケージの Setup フォルダー配下にある setup.exe をダブルクリックします。

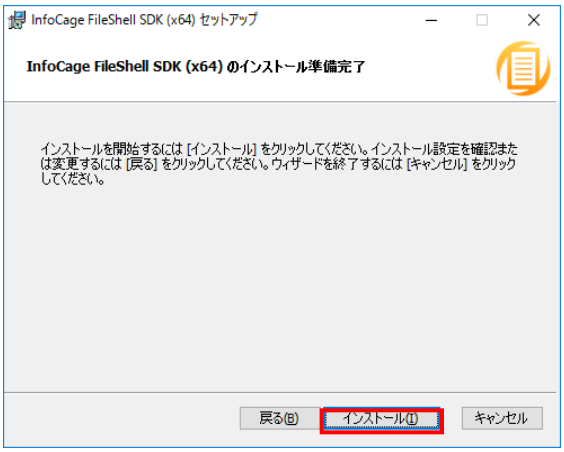
2. インストールウィザードが表示されます。[次へ]をクリックしてください。



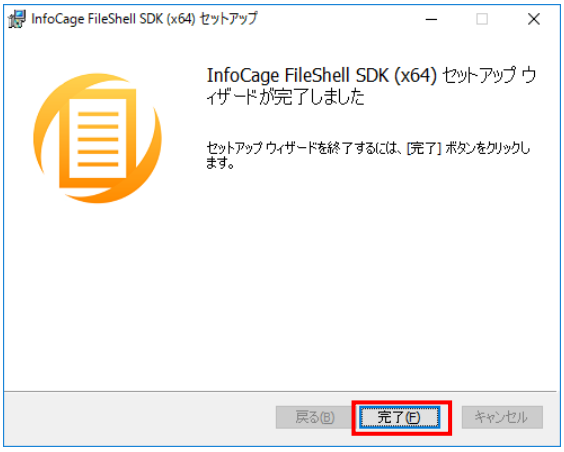
3. [インストール先のフォルダー]画面が表示されます。インストールフォルダーを指定し、[次へ]をクリックしてください。



4. [インストール]をクリックし、インストールを実行してください。



5. FileShell SDK のインストールが完了すると、以下の画面が表示されます。
[完了]をクリックして、インストールを完了してください。



以上で、FileShell SDK のインストールは、終了です。

4.5.4 MIP による分類/保護や解除をおこなう場合の設定

MIP による分類/保護や解除をおこなう場合は、FileShell SDK のインストール後に、以下の FileShell SDK のライブラリのコピーをおこなってください。

コピー元	コピーするライブラリ	コピー先
%FileShell SDK のインストールフォルダー%	IcfMsal.exe IcfMsal.exe.config Microsoft.Identity.Client.dll Microsoft.Identity.Client.xml mip_ClientTelemetry.dll mip_core.dll mip_file_sdk.dll mip_protection_sdk.dll mip_upe_sdk.dll	FileShell SDK の API を呼び出すアプリケーションの実行ファイル(exe)が存在するフォルダーに格納してください。 (例) API を呼び出すアプリケーションが、C:¥Program Files¥abc¥abc.exe の場合 C:¥Program Files¥abc
%FileShell SDK のインストールフォルダー%¥ja-JP	IcfMsal.resources.dll	FileShell SDK の API を呼び出すアプリケーションの実行ファイル(exe)が存在するフォルダー配下に ja-JP フォルダーを作成して、格納してください。 (例) API を呼び出すアプリケーションが、C:¥Program Files¥abc¥abc.exe の場合 C:¥Program Files¥abc¥ja-JP

＊ FileShell SDK は、デフォルトでは以下にインストールされます。

〈システムドライブ〉¥Program Files¥NEC¥InfoCageFileShellSDK¥

＊ FileShell SDK の API を呼び出すアプリケーションが、IIS のアプリケーションプールで動作するアプリケーションの場合は、FileShell SDK のライブラリを以下にコピーしてください。

〈システムドライブ〉¥Windows¥System32¥inetsrv
(IcfMsal.resources.dll は、〈システムドライブ〉¥Windows¥System32¥inetsrv¥ja-JP にコピーしてください)

FileShell SDK の API を呼び出すアプリケーションが、IIS のアプリケーション プールで動作するアプリケーションの場合は、タスクマネージャーのプロセスタブで"w3wp.exe"を終了させてからライブラリのコピーをおこなってください。

なお、タスクマネージャーのプロセス上に複数の"w3wp.exe"が存在する場合は、"w3wp.exe"のコマンドライン上に、開発中のアプリケーションのアプリケーション プール名が含まれるものを選択して終了してください(コマンドラインは、タスクマネージャーのプロセスタブの列に"コマンドライン"を追加することで確認できます。)

4.6 FileShell SDK の環境設定

FileShell SDK 環境設定ツールを起動し、以下の初期設定をおこないます。

参照 環境設定ツールの起動方法は「5.2 起動方法」を参照してください。

4.6.1 RMS 認証情報の設定

RMS 認証情報を設定します。

FileShell SDK を使用して OfficeIRM 形式／FileShell 形式でファイルを保護をするためには、RMS のライセンスサーバーおよび認証サーバーの URL を環境設定ツールにて設定する必要があります。

＊ NFP 形式のみを使用する場合はこの設定は必要ありません。

＊ OfficeIRM 形式／FileShell 形式で保護をするには RMS Client V2.1 がインストールされている必要があります。動作に必要なソフトウェアについては「1.3 動作環境について」および「4.2 必要なソフトウェアのインストール」を参照してください。

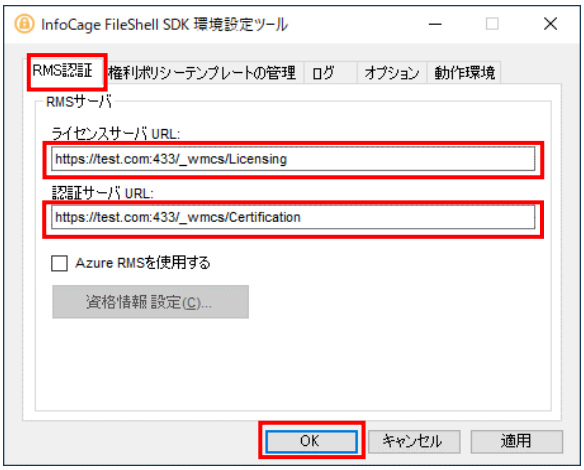
Operation

1. FileShell SDK インストール先の FileShellSDKSetting.exe を実行します。

   *FileShell SDK は、デフォルトでは以下にインストールされます。

   〈システムドライブ〉¥Program Files¥NEC¥InfoCageFileShellSDK¥

2. [RMS 認証]タブを選択し、RMS サーバーの[ライセンスサーバ URL]および[認証サーバ URL]にそれぞれの設定内容を入力後、[OK]をクリックします。



＊ [RMS 認証]タブでの詳細な設定については「5.3.1 RMS 認証情報設定」を参照してください。

4.6.2 ログ出力情報の設定

ログ出力情報を設定します。

参照 設定方法は「5.3.3 ログ出力情報設定」を参照してください。

4.7　権利ポリシーテンプレートの準備

FileShell SDK で権利ポリシーテンプレートによるファイル保護するためには、運用/使用する保護の形式に応じた権利ポリシーテンプレート(xml ファイル)が必要です。権利ポリシーテンプレートは、RMS サーバーや Azure RMS サーバー、もしくは FileShell サーバーおよび FileShell クライアント設定からエクスポートされたものを使用してください。

＊ MIP による分類/保護を利用する場合は、本節の操作は不要です。「4.10 ラベル ID の取得」の手順で使用するラベルのラベル ID を取得してください。

以下に権利ポリシーテンプレートの取得、および保存方法について、運用形態ごとに例をあげて説明します。

保存した権利ポリシーテンプレートやラベル ID は、「4.9 権利ポリシーテンプレートのインポート」で使用します。

4.7.1　オンプレミス RMS サーバー上の権利ポリシーテンプレートの取得と保存

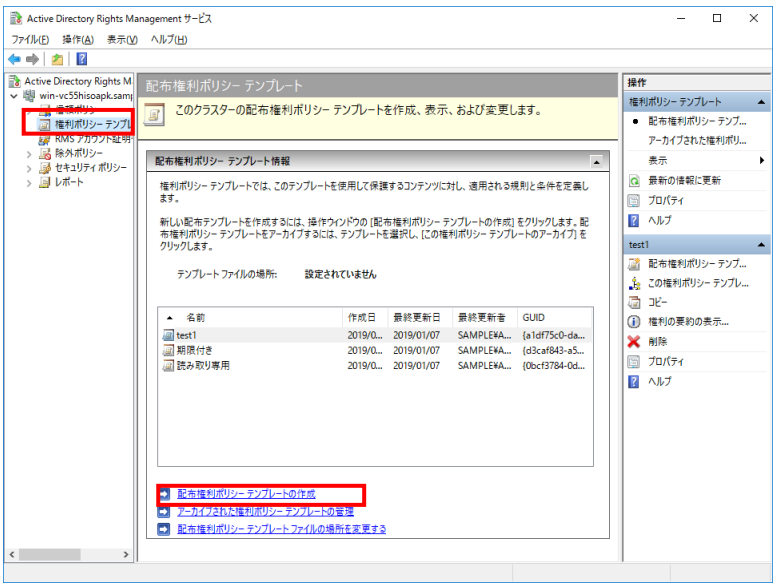
オンプレミスの RMS サーバーから権利ポリシーテンプレートを xml ファイル形式で保存する方法を説明します。

Notice

・FileShell SDK で指定する権利ポリシーテンプレートは Active Directory Rights Management サービス(AD RMS)サーバーで作成されたもの、もしくは、FileShell 製品が提供する権利ポリシー作成機能を利用して作成されたものでなければなりません。xml エディタを使って手動で編集をした場合等、他の方法で作成された権利ポリシーテンプレートを使用した場合の動作は保障されません。

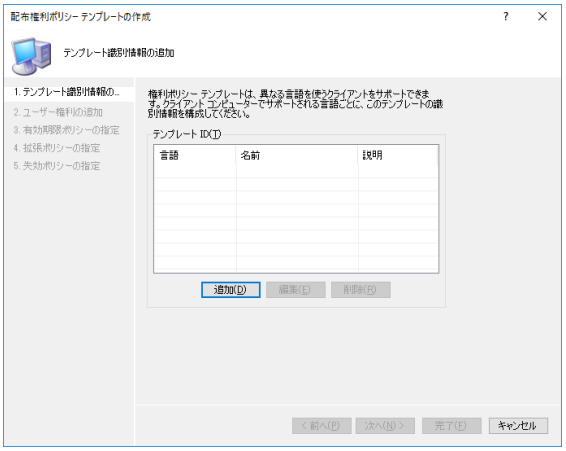
Operation

1. Active Directory Rights Management サービスの左部のツリーから「権利ポリシーテンプレート」を選択し、下部に表示される「配布権利ポリシーテンプレートの作成」を選択します。



2. 下記の配布権利ポリシーテンプレートの作成で表示されるダイアログに従い、テンプレート名、アクセスユーザーおよびアクセス権限、保護ファイルの有効期限等を必要に応じて指定して、権利ポリシー

テンプレートを作成します。作成した権利ポリシーテンプレートは「テンプレートファイルの場所」で指定されたパスに出力することができます。



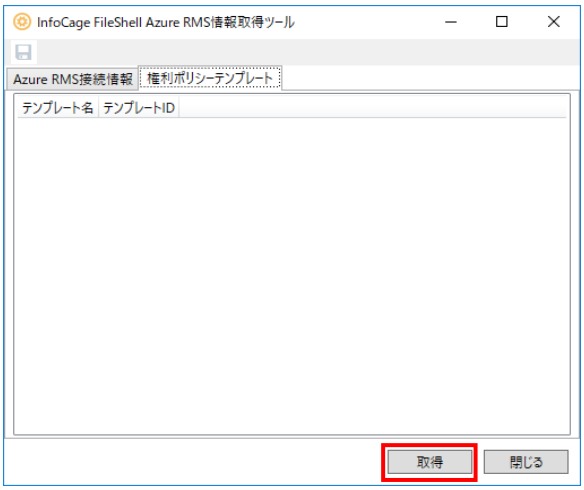
4.7.2 Azure RMS サーバー上の権利ポリシーテンプレートの取得と保存

「Azure RMS 接続情報取得ツール」を使用して、Azure RMS サーバー上の権利ポリシーテンプレートを xml ファイル形式で保存する方法を説明します。

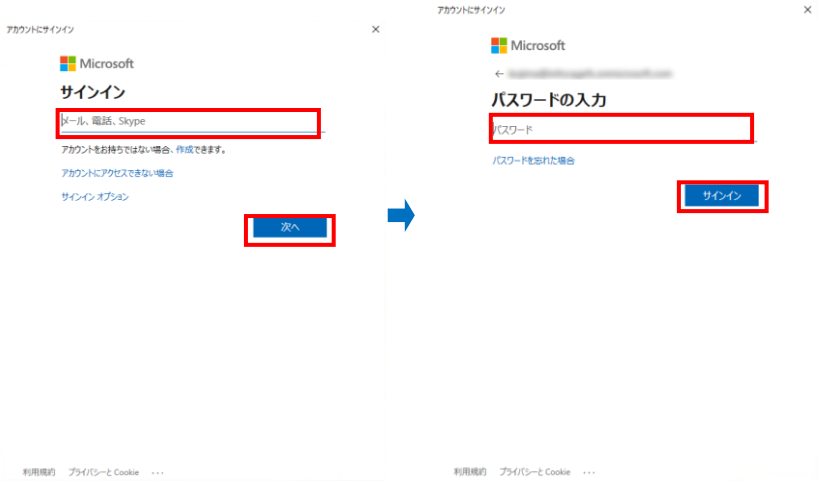
参照 「Azure RMS 接続情報取得ツール」のインストール方法については、『管理者ガイド』の「Azure RMS 接続情報取得ツールの導入手順」を参照してください。

Operation

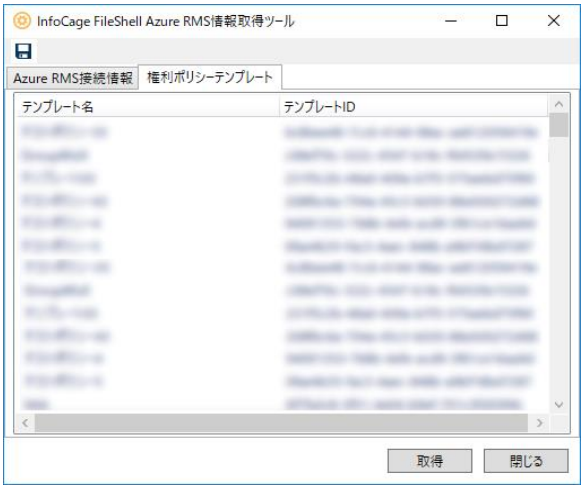
1. C:¥ IcfGetAzureInfo から、「IcfGetAzureInfoTool.exe」を起動します。
2. [権利ポリシーテンプレート]タブをクリックします。
3. テンプレート一覧画面が表示されます。
   [取得]をクリックします。



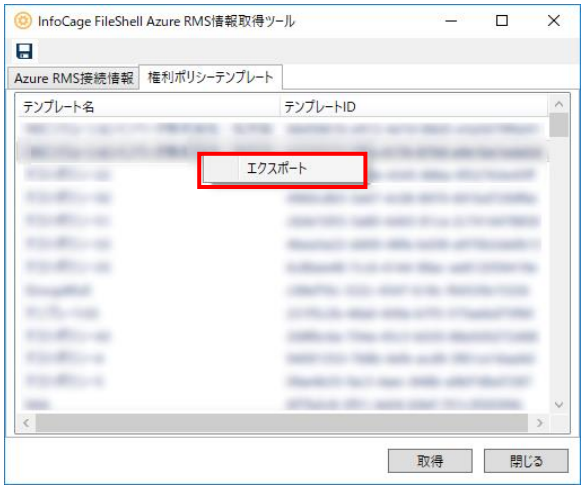
4. サインイン画面が表示された場合は、Azure のユーザーID とパスワードを入力します。



5. テンプレート一覧画面に Azure RMS サーバー上の権利ポリシーテンプレートの情報が表示されます。



6. 対象の権利ポリシーテンプレートを右クリックして、[エクスポート]を選択します。



7. 保存ダイアログでファイル名を入力し、[保存]をクリックします。

＊ 画面左上の は、権利ポリシーテンプレートの保存には使用できません。

以上で、Azure RMS サーバー上の権利ポリシーテンプレートの取得および保存は完了です。

4.7.3 NFP 権利ポリシーテンプレートの取得と保存

NFP で FileShell が運用されている環境で、FileShell サーバーから NFP 権利ポリシーテンプレートを xml ファイル形式で保存する方法を説明します。

＊ 利用する NFP 権利ポリシーテンプレートにより、取得・保存の方法が異なります。

＊ NFP 権利ポリシーテンプレートの詳細については、NEC File Protection Edition 管理者ガイド、もしくは NEC File Protection Edition 利用ガイドを参照してください。

4.7.3.1 サーバー認証版の NFP 権利ポリシーテンプレートを利用する場合

サーバー認証版の NFP 権利ポリシーテンプレートを利用する場合は、利用する NFP 権利ポリシーテンプレートを FileShell サーバーのバックアップアップ機能を用いて取得し、保存します。

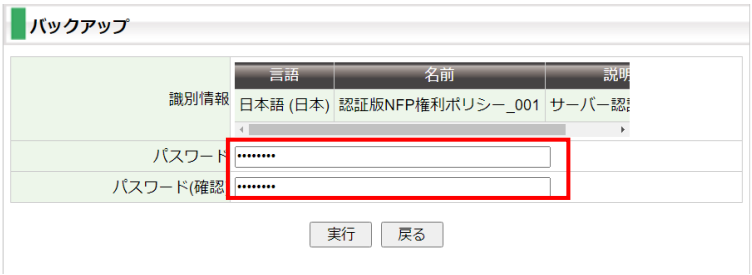
Operation

1. FileShell サーバーに全体管理者もしくは組織管理者でログインし、Web 管理コンソールの組織管理者用画面で[ラベル/権利ポリシ管理]を選択します。

2. 権利ポリシー一覧に表示されている「NFP 権利ポリシーテンプレート情報」から利用するサーバー認証版の NFP 権利ポリシーテンプレートの右部[操作]メニューで[バックアップ]を選択し[実行]をクリックします。

＊ サーバー認証版の NFP 権利ポリシーテンプレートは、名前の後ろに[認証版]と表示されています。

3. バックアップ画面が表示されます。インポートの際に使用するパスワードを設定し、[実行]をクリックすると、保存ダイアログが表示され NFP 権利ポリシーテンプレートを保存できます。



以上で、サーバー認証版の NFP 権利ポリシーテンプレートの取得および保存は終了です。

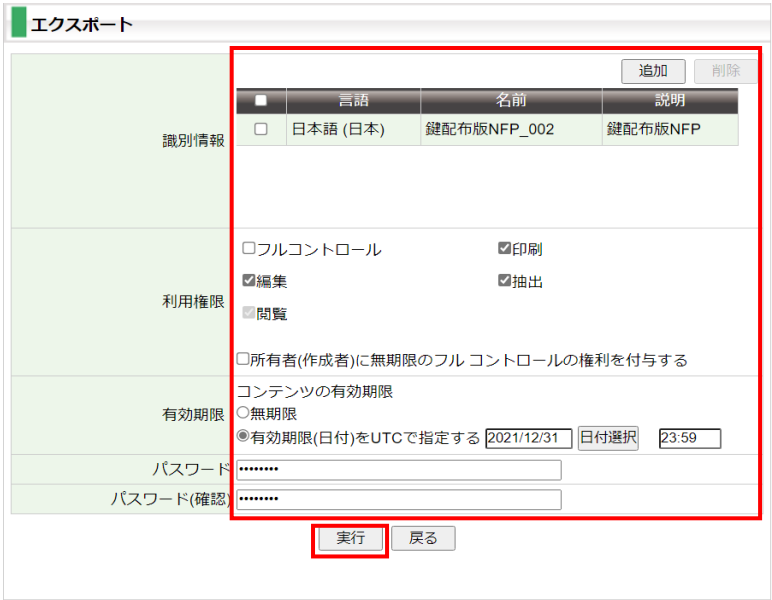
4.7.3.2 鍵配布版の NFP 権利ポリシーテンプレートを利用する場合

Operation

1. FileShell サーバーに全体管理者もしくは組織管理者でログインし、Web 管理コンソールの組織管理者用画面で「ラベル/権利ポリシ管理」を選択します。

2. 権利ポリシー一覧に表示されている「NFP 権利ポリシーテンプレート情報」から利用する鍵配布版の NFP 権利ポリシーテンプレートの右部[操作]メニューでエクスポートを選択し「実行」をクリックします。

3. エクスポート画面が表示されます。エクスポートする NFP 権利ポリシーの識別情報、付与する権利、有効期限、およびインポート時に使用するパスワードを設定し、[実行]ボタンをクリックすると、保存ダイアログが表示され NFP 権利ポリシーテンプレートを保存できます。



以上で、鍵配布版の NFP 権利ポリシーテンプレートの取得および保存は終了です。

4.8 Azure RMS 権利ポリシーテンプレートの編集

Azure RMS 環境で利用する権利ポリシーテンプレートを編集します。

＊ オンプレミス環境を利用する場合は本手順は不要です。

＊ MIP による分類/保護を利用する場合は、「4.11 ラベルへの保護解除権限の付与」を参照してください。

FileShell SDK でファイルを保護または保護解除する場合、ユーザーは

[アプリケーション(クライアント)ID]@[RightsManagementServiceId].rms.ap.aadrm.com となります。

このユーザーは Azure AD のユーザー管理画面から追加することはできません。

以下の手順で追加する必要があります。

Operation

1. Windows PowerShell から、本操作に必要となるモジュールをインストールします。
Windows の [スタートメニュー]からプログラムの一覧を表示し、[Windows PowerShell]の右クリックメニューから、「管理者として実行」を選択します。

2. 以下のコマンドを実行し、TLS1.2 を有効にします。

> [Net.ServicePointManager]::SecurityProtocol = [Net.ServicePointManager]::SecurityProtocol -bOR [Net.SecurityProtocolType]::Tls12

＊ お使いの環境が Windows Server 2019 以降の場合は、本コマンドの実行は不要です。

3. 以下のコマンドを実行し、次のモジュールをインストールします。

・ AIPService PowerShell モジュール

> Install-Module -Name AIPService

＊ お使いの環境に Azure Rights Management Administration Tool (AADRM)がインストールされている場合は、以下のコマンドを実行し、AADRM をアンインストールしてから AIPSevice PowerShell モジュールをインストールしてください。

> Uninstall-Module -Name AADRM

・ Exchange Online PowerShell モジュール

> Install-Module -Name ExchangeOnlineManagement

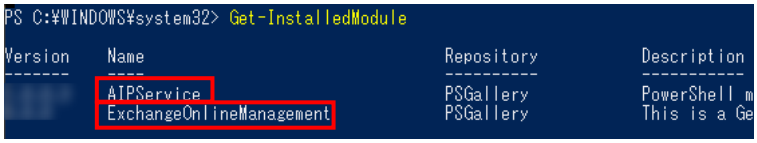
＊ パッケージマネージャー「nuget」のインストール確認メッセージが表示された場合は、「Y」を入力して続行してください。

＊ モジュールのインストール時に、「信頼されていないリポジトリからモジュールをインストールしようとしています。..」と表示された場合は、「Y」を入力して続行してください。

4. 以下のコマンドを実行し、モジュール一覧を取得します。

> Get-InstalledModule

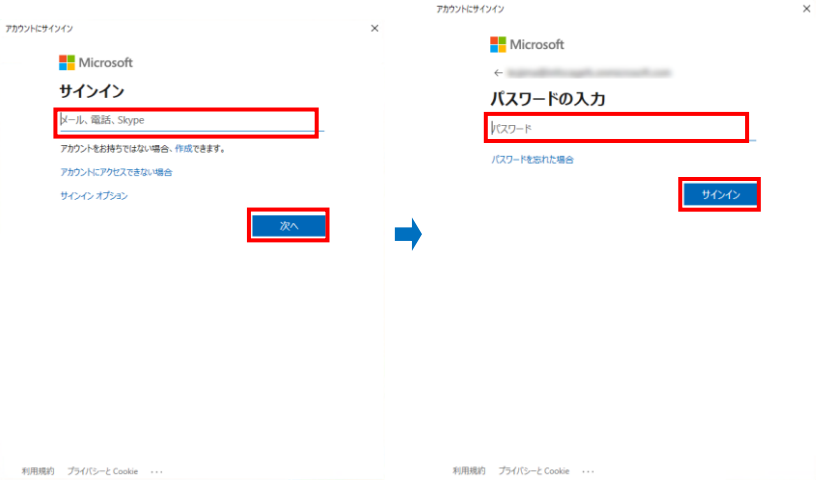
5. 一覧の「Name」に手順 3 でインストールしたモジュールが表示されていることを確認します。



6. 続いて、以下のコマンドを実行します。

> Connect-AIPService

7. 表示されたサインイン画面で Azure のユーザーID とパスワードを入力し、サインインします。

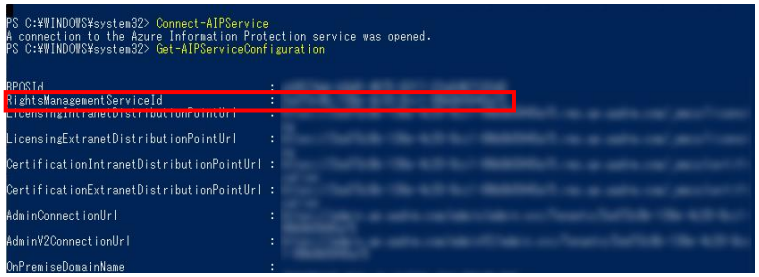


＊ グローバル管理者の資格情報を持つユーザーで接続してください。

8. 以下のコマンドを実行します。

> Get-AIPServiceConfiguration

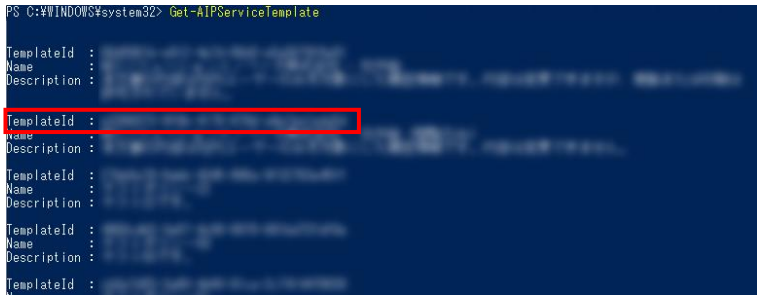
9. 「RightsManagementServiceId」の値を確認します。



10. 以下のコマンドを実行します。

> Get-AIPServiceTemplate

11. 編集する権利ポリシーテンプレートの「TemplateId」を確認します。



12. 以下のコマンドを実行し編集する権利ポリシーテンプレートをダウンロードします。

> Export-AIPServiceTemplate -templateid [TemplateId] -path [保存するファイル名]

例) c:¥Policy¥に sample.xml という名前で保存する場合

> Export-AIPServiceTemplate -templateid 01234567-89ab-cdef-0123-456789abcdef -path c:¥Policy¥sample.xml

13. 権利ポリシーテンプレートをメモ帳などで開き直接編集します。

ユーザーは[アプリケーション(クライアント)ID]@[RightsManagementServiceId].rms.ap.aadrm.com の形式で追加してください。

＊ アプリケーション(クライアント)ID]は、「4.4.1.1 Azure Portal でのアプリケーションの登録」の手順 **5** で取得したものを使用します。

14. 以下のコマンドを実行し編集した権利ポリシーテンプレートをアップロードします。

> Import-AIPServiceTemplate -path [アップロードするファイル名]

例) c:¥Policy¥sanmple.xml をアップロードする場合

> Import-AIPServiceTemplate -path c:¥Policy¥sample.xml

以上で、Azure RMS 権利ポリシーテンプレートの編集は終了です。

4.9 権利ポリシーテンプレートのインポート

FileShell SDK を使って開発したアプリケーションが利用する権利ポリシーテンプレートをインポートします。

＊ MIP による分類/保護を利用する場合は本節の手順は不要です。

＊ 下記説明文中の<SID>には、本ソフトウェアを動作させる時に使用するアカウントのものが入ります。
SID は、本ソフトウェアを動作させる時に使用するアカウントでログオンし、コマンドプロンプトから、「whoami /user」コマンドを実行します。
コマンドの詳細については、「whoami /?」を実行してください。

4.9.1 Office IRM/FileShell 形式で保護する場合

Office IRM/FileShell 形式での保護を利用する場合は、アプリケーションの設計に従い、以下の手順で保護時に利用する RMS の権利ポリシーテンプレートをインポートします。

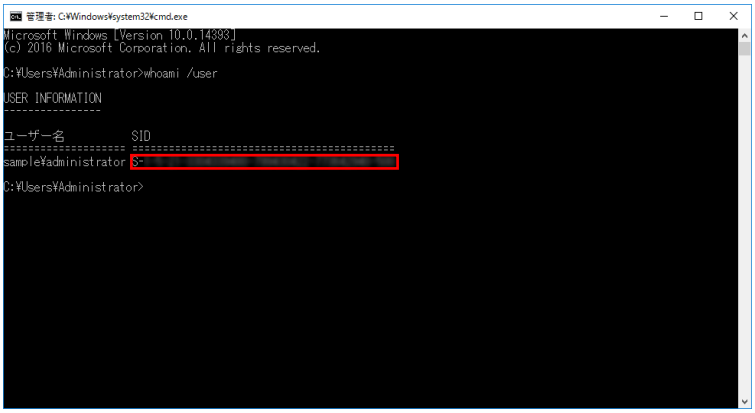
＊ Azure RMS サーバーで作成した権利ポリシーテンプレートを Unmanaged フォルダーに配置する場合、FileShell クライアントにて取得し、エクスポートしたものをご使用ください。
「4.8 Azure RMS 権利ポリシーテンプレートの編集」にて、PowerShell を用いて取得したものは使用できません。

＊ 「SYSTEM」、「LOCAL SERVICE」など、Windows のサービスアカウントで動作させる場合、手順 **2** で「FileShell SDK 環境設定ツール」を起動したあと、手順 **5** から実施してください。

Operation

1. 本ソフトウェアを動作させる時に使用するアカウントでログオンし、コマンドプロンプトから、「whoami /user」コマンドを実行して SID を取得します。

（取得した SID は手順 **4** で使用します。）



＊ SID は「S-」から始まるすべての文字列です。

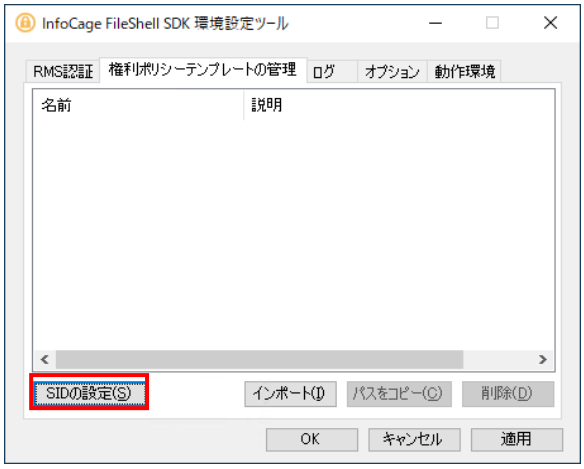
＊ コマンドの詳細については、「whoami /?」を実行してください

2. FileShell SDK インストール先の FileShellSDKSetting.exe を実行します。

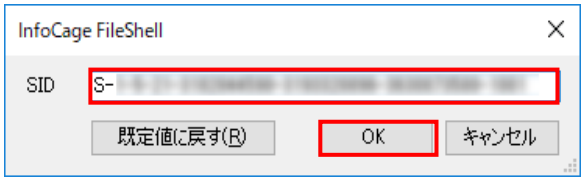
＊ デフォルトでは以下にインストールされています。

＊ <システムドライブ>¥Program Files¥NEC¥InfoCageFileShellSDK¥

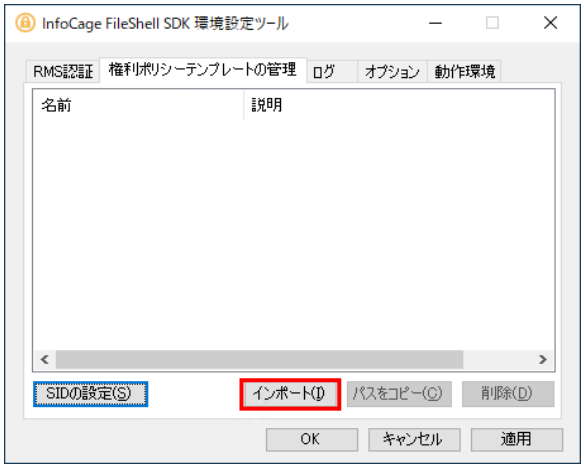
3. [権利ポリシーテンプレートの管理]のタブを選択し、[SID の設定]を選択します。



4. SID の設定画面が表示されます。入力欄に手順 **1** で確認した SID を入力し、OK をクリックします。



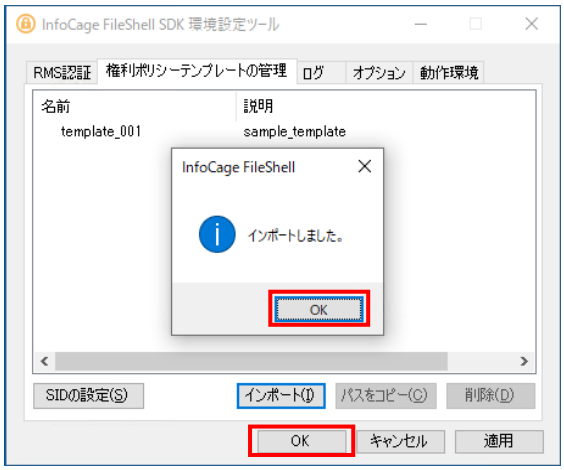
5. [権利ポリシーテンプレートの管理]のタブの[インポート]を選択します。



6. 任意のフォルダーに格納した権利ポリシーテンプレート(xml 形式)を選択し、インポートします。

＊ RMS の権利ポリシーテンプレートを使用する場合、権利ポリシーテンプレートは保護機能を使用するユーザーごとに SID を設定して、インポートする必要があります。SID を変更した場合は、権利ポリシーテンプレートをインポートしなおしてください。

7. インポートが完了すると、メッセージが表示され、一覧に権利ポリシーテンプレートが追加されます。メッセージの OK ボタンクリックした後、下部の OK ボタンで環境設定ツールを閉じます。



＊ 環境設定ツールでライセンスサーバーURL または認証サーバーURL の設定が空欄もしくは正しく入力されていない場合、[OK]／[適用]ボタンを押下した際にサーバーURL が不正である旨のエラーメッセージが表示され、設定操作を完了できません。本設定より先に「4.6.1 RMS 認証情報の設定」および「5.3.1 RMS 認証情報設定」を参考に設定を完了してください。(表示されるエラーの詳細については「5.3 環境設定ツールの操作」の注意書きを参照してください)。

以上で、Office IRM/FileShell 形式で保護する場合の権利ポリシーテンプレートのインポートは終了です。

4.9.2 NFP 形式で保護する場合

NFP 形式での保護を利用する場合は、アプリケーションの設計に従い、以下の手順で保護時に利用する NFP 権利ポリシーテンプレートをインポートします。

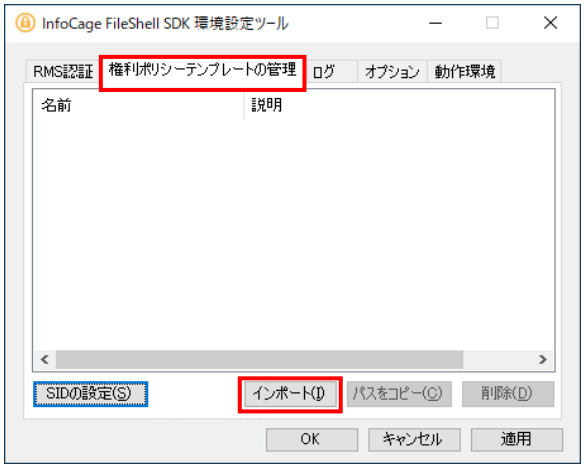
＊ 本ソフトウェアでは、パスワード暗号でエクスポート、もしくはバックアップされた NFP 権利ポリシーテンプレートを使用します。公開鍵暗号でエクスポートした NFP 権利ポリシーテンプレートはインポートできません。

Operation

1. FileShell SDK インストール先の FileShellSDKSetting.exe を実行します。

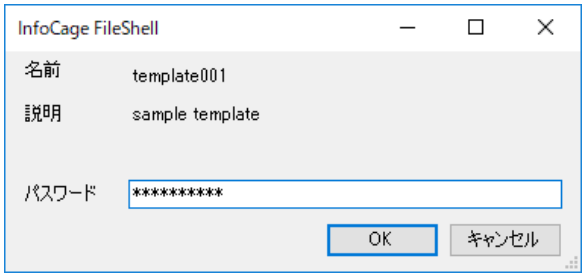
   ＊ デフォルトでは以下にインストールされています。
   〈システムドライブ〉¥Program Files¥NEC¥InfoCageFileShellSDK¥

2. [権利ポリシーテンプレートの管理]のタブを選択し、[インポート]を選択します。

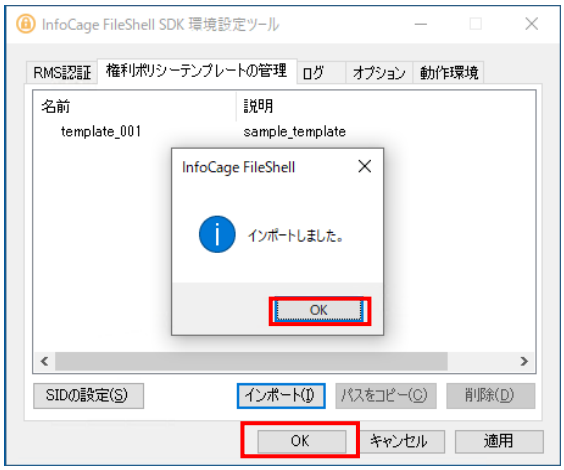


3. 任意のフォルダーに格納した NFP 権利ポリシーテンプレート(xml 形式)を選択し、インポートします。

   ＊ パスワード暗号でエクスポート、もしくはバックアップされた NFP 権利ポリシーテンプレート(xml 形式)をインポートするには、エクスポート時に設定したパスワードによる認証が必要です。パスワードについては NFP 権利ポリシーの管理者に確認してください。



4. インポートが完了すると、メッセージが表示され、一覧に権利ポリシーが追加されます。メッセージの OK ボタンクリックした後、下部の OK ボタンで環境設定ツールを閉じます。



＊ 権利ポリシーのインポート完了後に環境設定ツールの[OK]／[適用]ボタンを押下した際に、サーバーURL が不正である旨のエラーメッセージが表示された場合は、[キャンセル]ボタン、もしくは右上の「×」ボタンで環境設定ツールを終了してください（同エラーメッセージが表示されても、本操作による権利ポリシーの追加は完了しています）。

以上で、NFP 形式で保護する場合の権利ポリシーテンプレートのインポートは終了です。

4.10 ラベル ID の取得

FileShell SDK でラベルを使用する際に指定するラベル ID(Guid)を取得します。

Operation

1. Windows PowerShell から、本操作に必要となるモジュールをインストールします。
Windows の [スタートメニュー]からプログラムの一覧を表示し、[Windows PowerShell]の右クリックメニューから、「管理者として実行」を選択します。

2. 以下のコマンドを実行し、TLS1.2 を有効にします。

> [Net.ServicePointManager]::SecurityProtocol = [Net.ServicePointManager]::SecurityProtocol -bOR [Net.SecurityProtocolType]::Tls12

＊ お使いの環境が Windows Server 2019 以降の場合は、本コマンドの実行は不要です。

3. 以下のコマンドを実行し、次のモジュールをインストールします。

・ Exchange Online PowerShell モジュール

> Install-Module -Name ExchangeOnlineManagement

＊ パッケージマネージャー「nuget」のインストール確認メッセージが表示された場合は、「Y」を入力して続行してください。

＊ モジュールのインストール時に、「信頼されていないリポジトリからモジュールをインストールしようとしています。..」と表示された場合は、「Y」を入力して続行してください。

4. 以下のコマンドを実行し、モジュール一覧を取得します。

> Get-InstalledModule

5. 一覧の「Name」に手順 **3** でインストールしたモジュールが表示されていることを確認します。

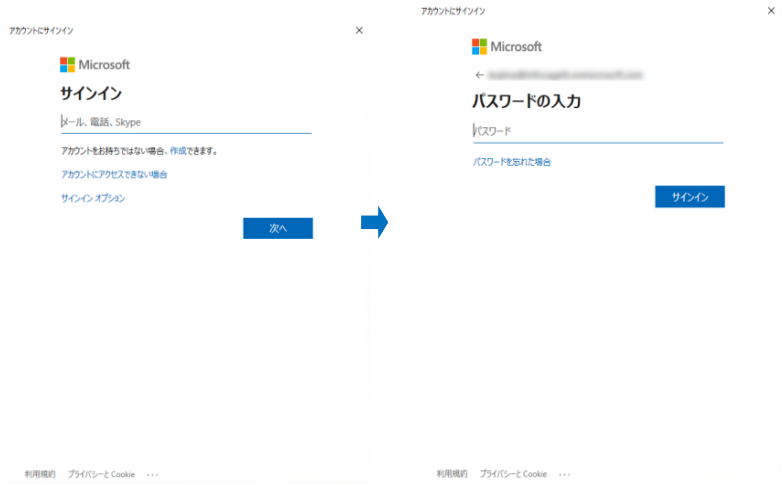
```
PS C:¥[redacted]> Get-InstalledModule

Version    Name                          Repository         Description
-------    ----                          ----------         -----------
[redacted] ExchangeOnlineManagement      PSGallery          This is a Gene
```

6. 以下のコマンドを実行します。

> Connect-IPPSSession

7. Microsoft のサインイン画面が表示された場合は、ユーザーID とパスワードを入力します。



＊ グローバル管理者の資格情報を持つユーザーで接続してください。

8. 以下のコマンドを実行します。

> Get-Label | Select-Object -Property DisplayName,Guid を実行します。

9. ラベルの DisplayName と Guid が表示されますので、使用するラベルの Guid を控えます。

```
PS C:¥[redacted]> Get-Label | Select-Object -Property DisplayName,Guid

DisplayName Guid
----------- ----
Label01     71a4bc1b-587b-464c-b44c-8b0bbabcbc5d
ラベル2     b348f528-ad77-434d-a293-78bc04746db3
Protect(... b93fa642-7ee3-47ad-8eb6-40d018f07f02
Label(Ad... 116dee22-e3c1-424d-9147-3044f961d9e0
```

10. 以下のコマンドを実行して、接続を切断します。

> Disconnect-ExchangeOnline

処理を続行してよいかを確認するメッセージが表示された場合は、「Y」を入力します。

以上で、ラベル ID の取得は終了です。

4.11 ラベルへの保護解除権限の付与

ラベルによって分類/保護されたファイルを FileShell SDK で解除する場合は、本節に記載の手順で FileShell SDK で使用するラベルに対して以下のユーザーを追加し、解除に必要な権限を付与してください。

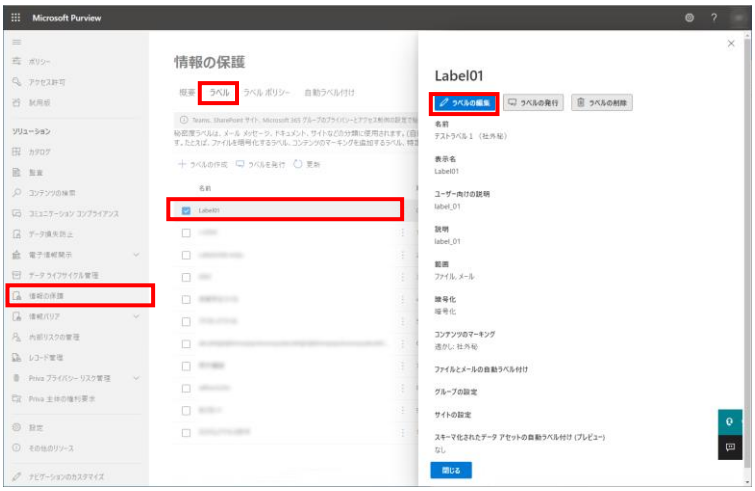
(追加するユーザー)

[アプリケーション(クライアント)ID]@[RightsManagementServiceId].rms.ap.aadrm.com

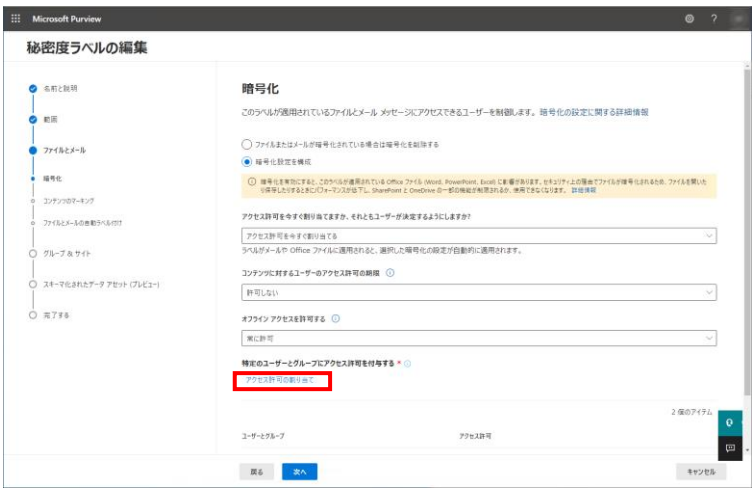
* 追加するユーザーの[アプリケーション(クライアント)ID]は、「4.4.1.1 Azure Portal でのアプリケーションの登録」の手順 *5* で取得したアプリケーション(クライアント)ID です。
* 追加するユーザーの[RightsManagementServiceId]は、「4.4.1.4 Azure RMS の接続情報の取得」で取得した RightsManagementServiceId です。
* FileShell SDK で解除したいすべてのラベルについて、上記ユーザーを追加してください。ただし設定に際しては、秘密度の高いラベルには本設定でのユーザーを追加しないなど、必要に応じた運用をおこなってください。
* アプリケーションを介さず、FileShell SDK 単体で分類/保護や解除をおこなう場合は、上記ユーザーを追加する必要はありません(この場合、FileShell SDK がラベルによって分類/保護されたファイルの「ファイルの所有者」となるため、ラベルの権限についての設定変更をおこなわなくても、解除権限を有しています)。
* 権利ポリシーによる保護と異なり、すでにラベルで分類/保護されたファイルに対しても、本節に記載の手順で上記ユーザーを追加することで解除できるようになります。過去の文書を FileShell SDK で解除したくない場合は、FileShell SDK 用のラベルを別途新規作成するなど、必要に応じた運用をおこなってください。

Operation

1. Microsoft Purview コンプライアンス ポータルの[情報の保護]－[ラベル]で、利用するラベルを選択し、[ラベルの編集]をクリックします。



2. 秘密度ラベルの編集画面が表示されますので、画面下部の[次へ]をクリックして[暗号化]の設定画面まで進め、[特定のユーザーとグループにアクセス許可を付与する]の[アクセス許可の割り当て]をクリックします。



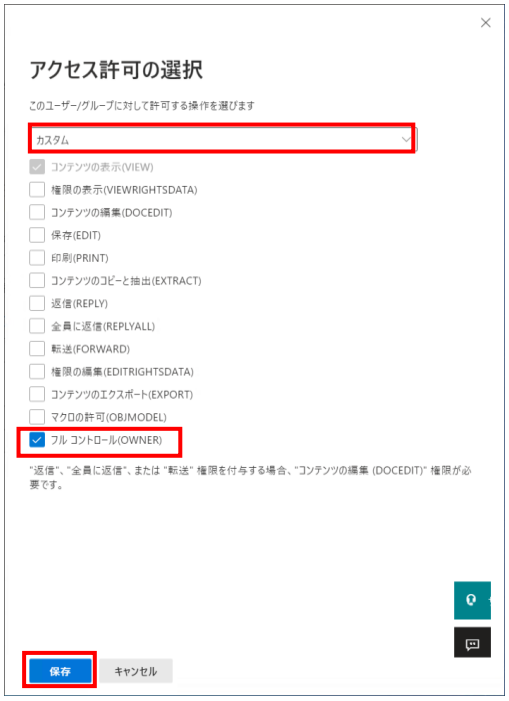
3. アクセス許可の割り当てで、「特定のメールアドレスまたはドメインを追加する」を選択し、以下のメールアドレスを追加します。

[アプリケーション(クライアント)ID] @[RightsManagementServiceId].rms.ap.aadrm.com

＊ ラベルにグループメーリングリストが設定されている場合は、アクティブディレクトリのように そのグループメーリングリストを追加することもできます。

4. 手順 **3** で設定したメールアドレスが追加されていることを確認し、[アクセス許可の選択]をクリックします。

5. 「アクセス許可の選択」でアクセス許可の設定をおこないます。[カスタム]を選択し、[フルコントール(OWNER)]にチェックを入れて、[保存]をクリックします。



6. 手順 **5** で設定したアクセス許可が設定されていることを確認し、[保存]をクリックします。

7. 画面下部の[次へ]をクリックして[設定を確認して完了]の画面まで進め、内容を確認後、[ラベルの保存]をクリックします。

8. ラベルの更新が完了した旨の画面が表示されますので、[完了]をクリックします。

以上で、FileShell SDK で使用するラベルへの保護解除権限の付与は終了です。

第5章 環境設定ツール

FileShell SDK を利用するためには RMS 認証やログなどの設定をおこなう必要があります。
本章では、それらの設定をおこなうための環境設定ツールの利用方法を記載します。

5.1 機能一覧

FileShell SDK 環境設定ツールでは、以下の機能を提供します。

機能名	概要
RMS 認証情報設定	RMS サーバーの認証に必要な情報を設定します。
権利ポリシー設定	権利ポリシーテンプレートをインポートし、使用する権利ポリシーのパスを取得します。
ログ出力情報設定	ログファイル出力に必要な情報を設定します。
オプション設定	動作に関する、その他の設定をおこないます。
動作環境表示	FileShell SDK のモジュール情報を表示します。

5.2 起動方法

FileShell SDK インストール先の FileShellSDKSetting.exe を実行します。

＊ デフォルトでは以下にインストールされています。
〈システムドライブ〉¥Program Files¥NEC¥InfoCageFileShellSDK¥

5.3 環境設定ツールの操作

5.3.1 RMS 認証情報設定

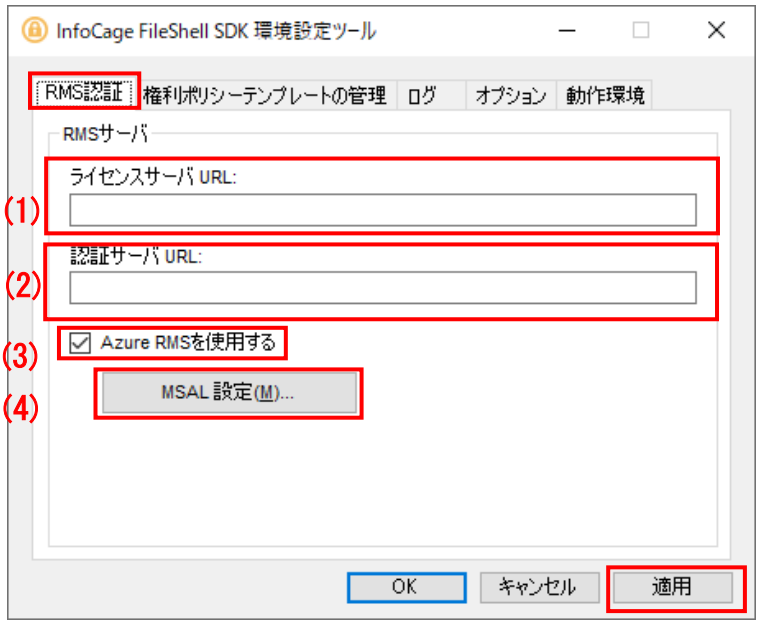
RMS の認証情報の設定は、FileShell SDK インストール時の初期設定をする場合、あるいは、RMS サーバー情報を変更する場合におこないます。

＊ NFP 形式のみを使用する場合は、この設定は必要ありません。

＊ NFP 形式のみを使用する場合は、FileShell での保護に RMS Client V2.1 は不要です。他の用途で必要なければ削除されることを推奨します。

Operation

1. [RMS 認証]画面の各項目を指定して、[適用]をクリックします。



項目		内容
(1)	ライセンスサーバ URL	オンプレミスの RMS サーバーを利用する場合は、RMS ライセンスサーバーの URL を指定します。 例) https://test.com/_wmcs/licensing Azure RMS を利用する場合は、「4.4.1.4 Azure RMS の接続情報の取得」で取得した"LicensingIntranetDistributionPointUrl"を指定します。 例) https://aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee.rms.ap.aadrm.com/_wmcs/licensing ＊ MIP を利用する場合は設定不要です。 ＊ NFP のみを利用する場合は設定不要です。
(2)	認証サーバ URL	オンプレミスの RMS サーバーを利用する場合は、RMS 認証(証明)サーバーの URL を指定します。 例) https://test.com/_wmcs/certification Azure RMS を利用する場合は、「4.4.1.4 Azure RMS の接続情報の取得」で取得した"CertificationIntranetDistributionPointUrl"を指定します。 例) https://aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee.rms.ap.aadrm.com/_wmcs/certification ＊ MIP を利用する場合は設定不要です。 ＊ NFP のみを利用する場合は設定不要です。
(3)	[Azure RMS を使用する]チェックボックス	MIP、もしくは Azure RMS を使用する場合にチェックします。 チェックボックスを ON にすると[MSAL 設定]ボタンが有効になります。
(4)	[MSAL 設定]ボタン	[MSAL 設定]画面を表示します。

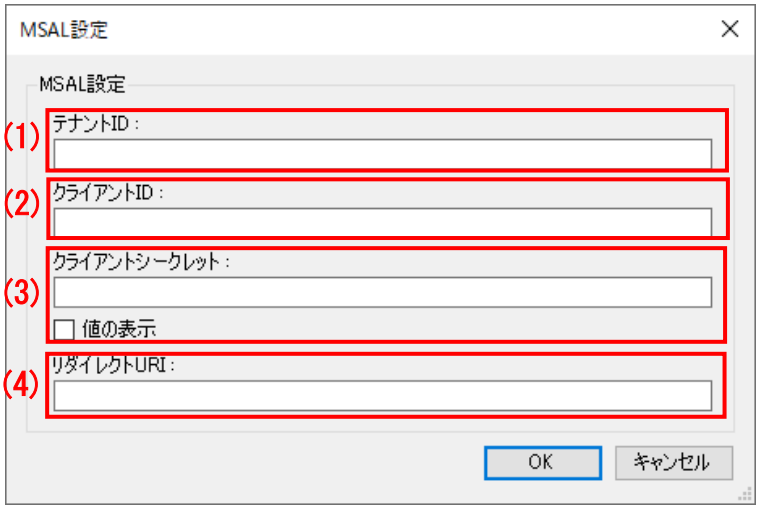
＊ ポート番号を含め、RMS サーバーの管理画面で表示されているのと同じ URL を指定してください。

＊ Azure RMS に接続するためにプロキシサーバーの設定が必要なネットワーク環境の場合、FileShell SDK を利用するアプリケーションの実行アカウントにプロキシサーバーの設定が適用されている必要があります。「2.1 運用上の注意事項」を参考に、設定をおこなってください、

2. MIP、もしくは Azure RMS を使用する場合は、[MSAL 設定]画面の各項目を指定して、[OK]をクリックします。

Notice

MSAL 設定で必要となる、クライアントシークレットは、有効期限が切れた場合、Azure RMS の認証がおこなえなくなるため、運用年数、セキュリティリスクなどを考慮の上、適切な期限を設定し、運用中に期限が切れることのないよう管理をおこなってください。
有効期限が切れた場合は、「4.4.1.3 クライアントシークレットの作成」に記載の手順でクライアントシークレットを再作成し、適用しなおしてください。

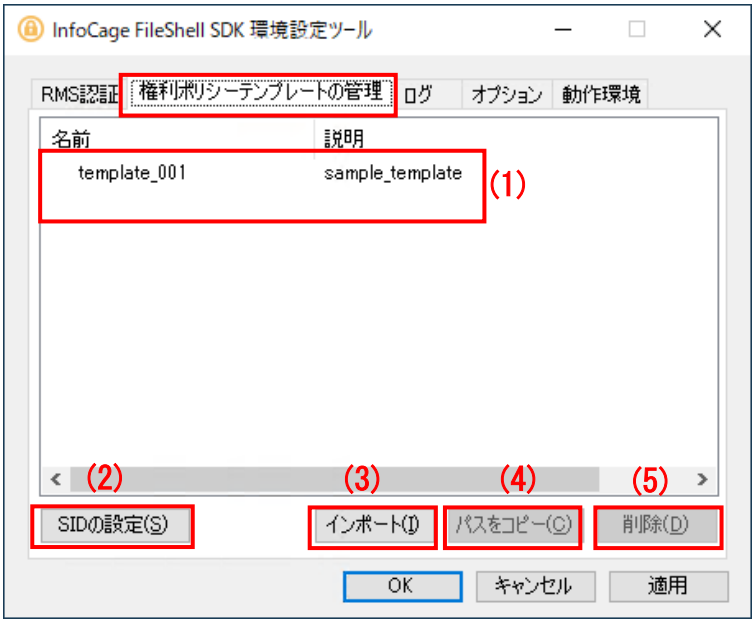


項目		内容
(1)	テナント ID	「4.4.1.1 Azure Portal でのアプリケーションの登録」で取得した「ディレクトリ(テナント)ID)」を指定します。
(2)	クライアント ID	「4.4.1.1 Azure Portal でのアプリケーションの登録」で設定した「アプリケーション(クライアント)ID」を指定します。
(3)	クライアントシークレット	「4.4.1.3 クライアントシークレットの作成」で作成したクライアントシークレットの値を指定します。 入力した値を確認したい場合は、[値の表示]にチェックを入れます。 ＊ FileShell SDK V6.1 より、Azure RMS を利用して保護/保護解除をおこなう場合の対称鍵(Symmetric Key) による認証はできなくなりました。 V6.1 未満の SDK からアップデートする場合など、従来の環境から引き続き Azure RMS を使用する場合でも、「4.4.1 MIP/Azure RMS を利用する場合に必要な情報の設定・取得」に記載の手順で Azure Portal にてアプリケーション登録～クライアントシークレット作成をおこない、クライアントシークレットの値を指定してください。
(4)	リダイレクト URI	「4.4.1.1 Azure Portal でのアプリケーションの登録」で設定したリダイレクト URI を指定します。

＊ 本設定は、IcfsProtector.dll のロード時に読み込まれます。設定を変更した場合は、IcfsProtector.dll の再ロードが必要です。

5.3.2 権利ポリシーテンプレートの管理

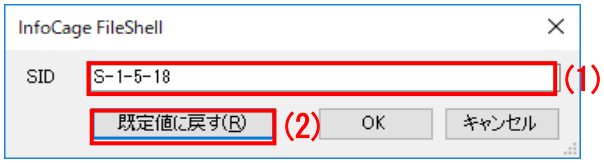
権利ポリシーテンプレートの管理は、FileShell SDK を使って開発したアプリケーションが利用する権利ポリシーテンプレートのインポートや削除などの管理をする場合におこないます。



項目		内容
(1)	権利ポリシーテンプレートの一覧	現在選択している SID にインポートされている権利ポリシーテンプレートを表示します。
(2)	SID の設定	権利ポリシーテンプレートを利用したいアカウントの SID を設定します。 ＊ NFP 形式のみを利用する場合は使用しません。 また、RMS Client V2.1 がインストールされていない環境では、このボタンは表示されません。 ＊ Office IRM/FileShell 形式の保護を使用する場合、権利ポリシーテンプレートは SDK を使用するアカウントごとに SID を設定して、インポートする必要があります。SID を変更した場合は、権利ポリシーテンプレートをその SID に対してインポートしなおしてください。
(3)	インポート	権利ポリシーテンプレート(xml ファイル)を選択し、インポートします。 ＊ SID が既定値以外の場合は、インポート前に SID の変更をおこなってください。
(4)	パスをコピー	選択した権利ポリシーテンプレートの配置パスを取得します。 ＊ 取得したパスはクリップボードにコピーされます。コピーされた内容や、メッセージは表示されません。
(5)	削除	選択した権利ポリシーテンプレートを削除します。

5.3.2.1　SID の設定

権利ポリシーテンプレートを使用するプロセスの実行アカウントの SID を設定します。



項目		内容
(1)	SID	権利ポリシーテンプレートに関連づける SID を指定します。 適用するには、本欄に入力後、OK をクリックします。
(2)	既定値に戻す	SID を既定値に戻します。 ＊ 既定値は S-1-5-18(LocalSystem)です。

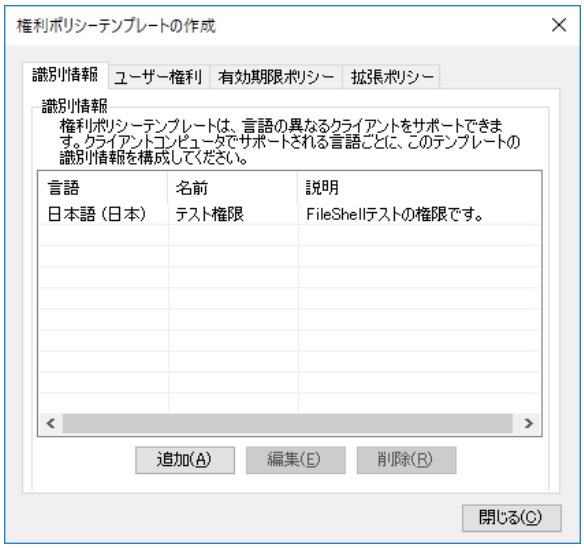
5.3.2.2　インポートされている権利ポリシーテンプレートの確認

権利ポリシーの一覧に表示されている権利ポリシーテンプレートをダブルクリックすることで、インポートされている権利ポリシーテンプレートの内容を確認します。

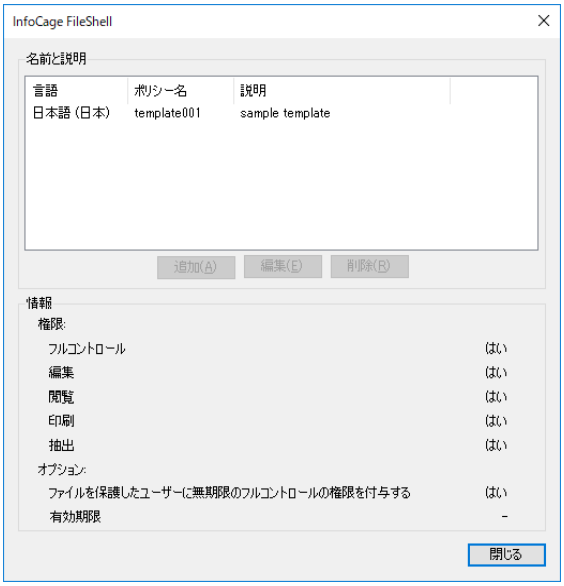
表示される内容は、RMS の権利ポリシーテンプレート、NFP 権利ポリシーテンプレートでそれぞれ異なります。

＊　環境設定ツールでは、権利ポリシーテンプレートの内容を編集することはできません。
編集、削除等の操作をおこなっても、内容は保存されません

● **RMS の権利ポリシーテンプレートの場合**



● NFP 権利ポリシーテンプレートの場合



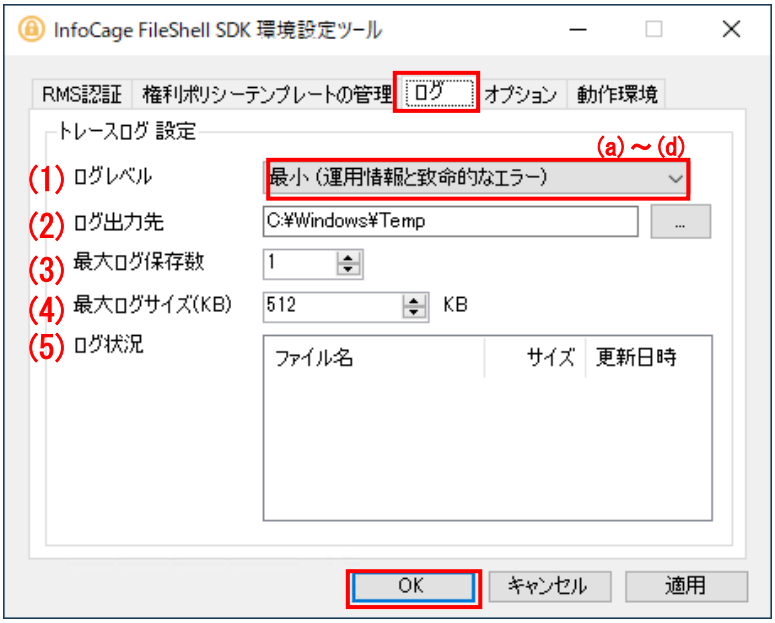
5.3.3 ログ出力情報設定

ログ出力情報の設定は、運用のためのログ設定をおこなう場合や FileShell SDK で何か問題が発生した場合等、SDK のログレベルやサイズを変更したい場合におこないます。

本設定により出力されるログは、FileShell SDK で障害が発生した場合等に、製品サポート側での解析で利用します。

Operation

[ログ]ページの各項目を指定して、[OK]をクリックします。



項目		内容
(1)	ログレベル	ログのレベルを指定します。指定できるレベルは以下のとおりです。 (a)最小(運用情報と致命的なエラー) ・・・運用情報と致命的なエラーを出力します (b)中(一般エラー) ・・・(a)に加えて継続可能なエラーを出力します (c)大(警告) ・・・(b)に加えて重要な動作情報を出力します (d)詳細(デバッグ情報) ・・・(c)に加えて詳細な動作情報を出力します 既定値: (a)最小(運用情報と致命的なエラー)
(2)	ログ出力先	[参照]ボタンを選択し[フォルダーの参照]ダイアログよりログの出力先のディレクトリを選択します。本フォルダーへのアクセス権限は、FileShell SDK の実行アカウントに対し変更権限以上が必要です。 既定値:Windows ディレクトリの Temp
(3)	最大ログ保存数	最大ログサイズを超えた場合に、ローテートするログファイルの最大個数を指定します。(1～20)　既定値:1
(4)	最大ログサイズ(KB)	1 つのログファイルの最大サイズを KB 単位で指定します。(5122,097,151～(512K～2G))　既定値:512
(5)	ログ状況	ログの出力先フォルダーに出力されているログファイルの情報を表示します。表示される情報は、ファイル名、サイズ、更新日時です。

5.3.4 オプション設定

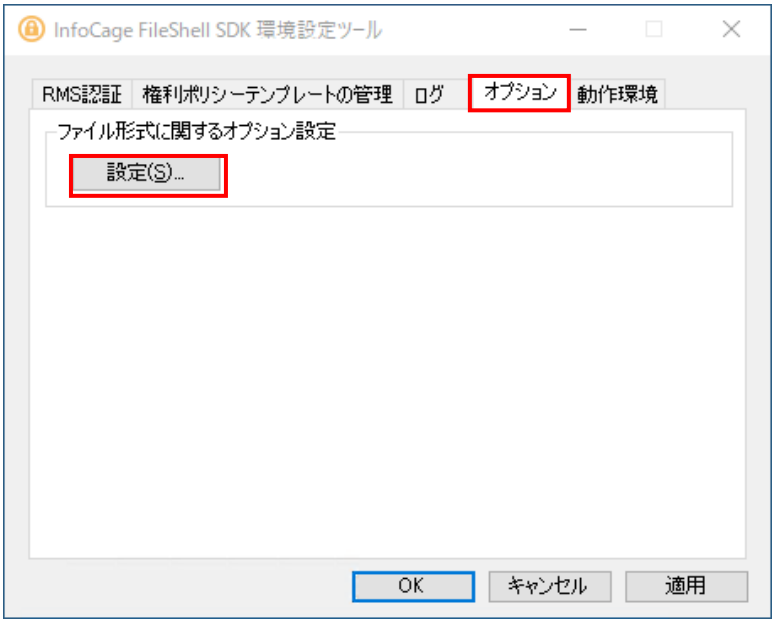
動作に関する、その他の設定をおこないます。

5.3.4.1 ファイル形式に関するオプション設定

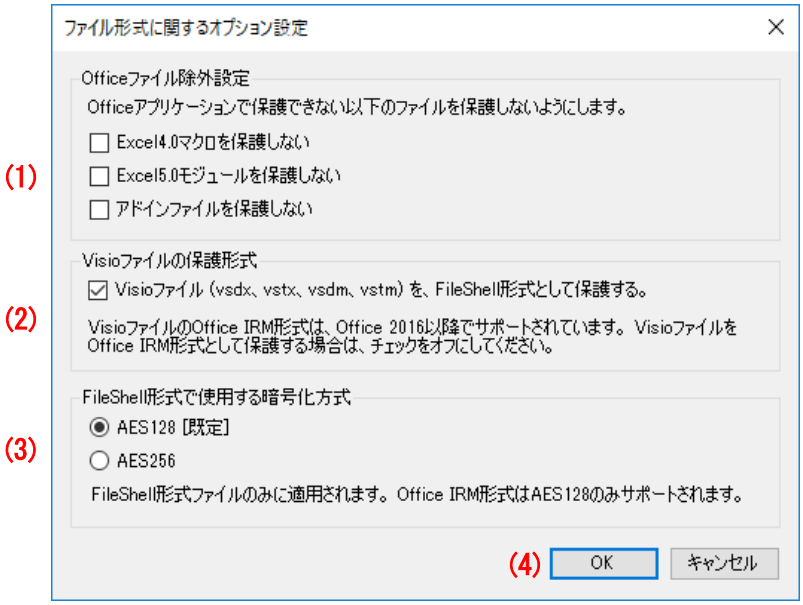
Office ファイルの除外設定および FileShell 形式の保護における暗号化方式の設定をおこないます。

＊ NFP 形式のみを使用する場合は、この設定は必要ありません。

[ファイル形式に関するオプション設定]の「設定」をクリックしてください。



「ファイル形式に関するオプション設定」画面



項目		内容
(1)	Office ファイル除外設定	
	[Excel4.0 マクロを保護しない]チェックボックス	Office アプリケーションで保護できない形式のファイルを、FileShell でも保護しないようにする場合に設定します。 ＊ 保護しないように指定した場合でも、以下の形式のファイルは保護されます。 ・ 読み取りパスワード付き Excel2007 形式 Excel アドインファイル
	[Excel5.0 モジュールを保護しない] チェックボックス	
	[アドインファイルを保護しない] チェックボックス	
(2)	Visio ファイルの保護形式	
	[Visio ファイル(vsdx、vstx、vsdm、vstm)を FileShell 形式として保護する]チェックボックス	Microsoft Visio のファイル(vsdx、vstx、vsdm、vstm)の保護に FileShell 形式を使用する場合にチェックを入れます。 ＊ チェックを外すと、Visio ファイルの保護に Office IRM 形式を使用します。 ＊ 既定は、チェック入(Visio ファイルを FileShell 形式として保護する」です。
(3)	FileShell 形式で使用する暗号化方式	
	[AES128]ラジオボタン	FileShell 形式での保護(暗号化)に、AES128 を使用します(既定)。
	[AES256]ラジオボタン	FileShell 形式での保護(暗号化)に、AES256 を使用します。
(4)	[OK]ボタン	ファイル形式に関するオプションを設定し「オプション設定」画面に戻ります。
	[キャンセル]ボタン	キャンセルします。

＊ 「FileShell 形式で使用する暗号化方式」の設定が AES256 を使用する設定の場合でも、OfficeIRM 形式では、AES128 で保護(暗号化)されます。

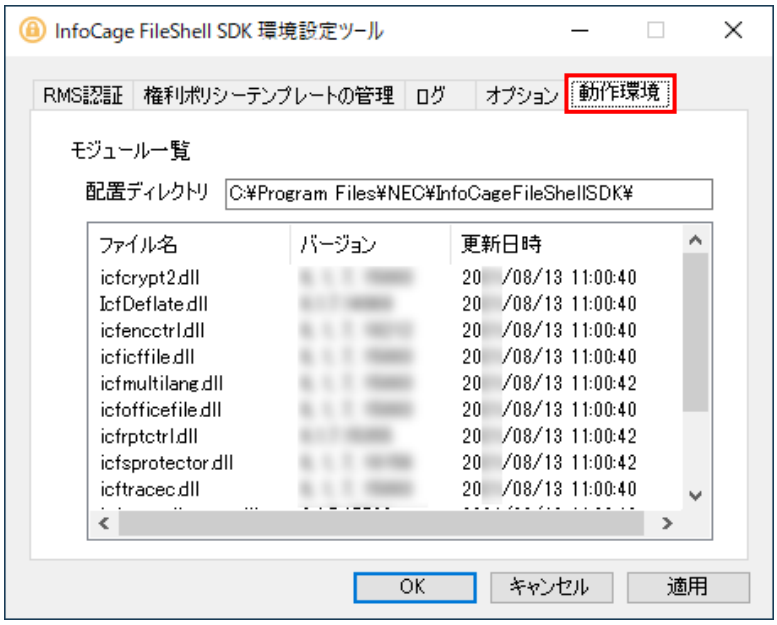
＊ 「FileShell 形式で使用する暗号化方式」の設定により、AES256 を使用して保護されたファイルは、FileShell クライアント Ver.3.0 以前の環境で開くことはできますが、保存することはできません。(ただし、Ver.3.0.291.9561 以降は、保存も可能となります。)
なお、FileShell クライアント Ver.3.1 以降では、保護されたファイルが AES128、AES256 で混在している場合でも読み書きすることができます。

5.3.5 動作環境表示

FileShell SDK の動作環境情報を表示します。

動作環境情報の表示は、FileShell SDK が正常に動作しない場合等、FileShell SDK のモジュール情報を確認したい場合におこないます。

＊ 本画面では、モジュール情報の確認のみ可能です、この画面で設定の変更などをおこなうことはできません。



5.3.6 設定情報ファイル出力

環境設定ツールの[OK]ボタン、または、[適用]ボタンをクリックしたタイミングで、環境設定ツールで設定した情報や FileShell SDK のモジュールバージョンの一覧が記載された設定情報ファイルが出力されます。

このファイルは、障害が発生し製品のサポート窓口に FileShell SDK の設定情報を送付しなければならない場合等に利用します。

《ファイル名》

IcfsProtector.env

《出力先》

ログ出力先フォルダー

《形式》

```
[Environment]
        動作環境情報

[RMS Authenticate]
        RMS 認証情報

[Log]
        ログ出力設定情報
```

第6章 バージョンアップインストール

FileShell SDK をバージョンアップインストールする手順を説明します。

Notice

・FileShell SDK V6.3 より、FileShell SDK で作成するアプリケーションの開始時と終了時に、それぞれ API による初期化処理、および終了処理をおこなう必要があります。
これにともない、V6.3 未満の環境からアップデートする場合は、V6.3 未満の FileShell SDK で作成したアプリケーションに対して、「3.3.1 FileShell SDK の初期化処理 ― IcfsInitialize()」、および「3.3.13 FileShell SDK の終了処理 ― IcfsUninitialize()」を追加してアプリケーションを更新する必要があります。

＊「3.5 サンプルコード」にこれらの処理を追加した例を掲載しておりますので、あわせてご確認ください。

FileShell SDK V6.3 環境下で上記の初期化、および終了処理を追加していないアプリケーションを実行すると、アプリケーションがフリーズするなどの事象が発生する場合があります。アプリケーションを更新しない場合は、V6.2 以下の FileShell SDK をご利用ください。

なお、FileShell SDK V6.3 を利用しない環境では以下の事象が発生する場合があります。
- AES 256-CBC で保護された、Office ファイルが保護解除できない。
- 「5.3.1 RMS 認証情報設定」の[MSAL 設定]に設定情報がない場合に、PDFv2 形式ファイルが二重保護される。

・FileShell SDK V6.1 より、Azure RMS を利用して保護/保護解除をおこなう場合、対称鍵(Symmetric Key) による認証はできなくなりました。これにともない、FileShell SDK V6.1 未満の環境からアップデートする場合は、Azure RMS の認証をクライアントシークレットによる認証に変更する必要があります。
FileShell SDK のバージョンアップをおこなう前に、「4.4.1 MIP/Azure RMS を利用する場合に必要な情報の設定・取得」に記載の手順で、Azure Portal にてアプリケーションを登録し、クライアントシークレットの作成をおこなってください。なお、FileShell SDK のバージョンアップ後に、「5.3 環境設定ツールの操作」の「5.3.1 RMS 認証情報設定」を参照し、MIP/Azure RMS の設定をおこなってください。

6.1 V2.0 以上からのバージョンアップ

インストーラを実行し、FileShell SDK モジュールのバージョンアップをおこなってください。

参照 FileShell SDK モジュールのバージョンアップ方法はインストールと同様です。

「4.5.3 インストール」を参照してください。

6.2 V2.0 未満からのバージョンアップ

FileShell SDK の V2.0 未満からのバージョンアップはできません。FileShell SDK の V2.0 未満を導入済みの場合、あらかじめ、そのバージョンの利用ガイドに従ってアンインストールする必要があります。

Notice

・本バージョンの FileShell SDK では、V1.1.5 未満の FileShell SDK 用に作成したアプリケーションは動作しません。
・本バージョンの FileShell SDK では、V1.1.5 未満の FileShell SDK と同じマシン上にインストールして動作させることはできません。

Operation

1. FileShell SDK を、そのバージョンの利用ガイドに従ってアンインストールします。

   新バージョンを同じ設定でインストールする場合は、アンインストール前に、環境設定ツールの情報を記録しておいてください。特別にレジストリ変更等をおこなっている場合、その情報は個別に記録しておく必要があります。

2. サーバーを再起動します。

3. 本バージョンの FileShell SDK を、本書の手順に従って新規インストールします。

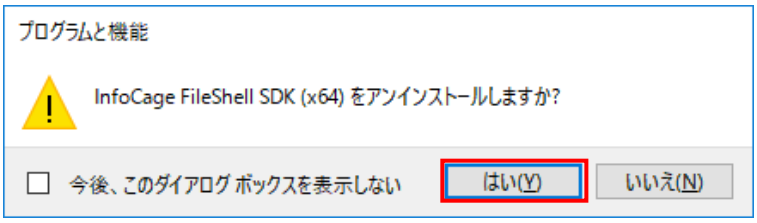
第7章 アンインストール

FileShell SDK をアンインストールする手順を説明します。

Operation

1. Windows のスタートメニューから、[コントロールパネル]-[プログラムと機能]を選択してください。
2. インストールされているプログラムの一覧が表示されます。
   [InfoCage FileShell SDK]を選択して、[アンインストール]をクリックしてください。

3. アンインストールの確認画面が表示されます。[はい]をクリックしてください。



＊ FileShell SDK のアンインストール後に、OS の再起動の必要はありません。

以上で、FileShell SDK のアンインストールは、終了です。

InfoCage FileShell SDK 利用ガイド
Ver 6.3

NEC ソリューションイノベータ株式会社

東京都江東区新木場一丁目 18 番 7 号
TEL(03)5534-2222 (代)

Copyright© NEC Solution Innovators, Ltd. 2021-2023.

NEC ソリューションイノベータ株式会社の許可なく複製・改変等を行うことはできません